

PATENT ABSTRACTS OF JAPAN

(11) Publication number : 08-263454

(43) Date of publication of application : 11.10.1996

(51) Int.CI.

G06F 15/16

(21) Application number : 07-090314

(71) Applicant : HITACHI LTD

(22) Date of filing : 23.03.1995

(72) Inventor : NODA TADASHI

YOSHIOKA SEIICHIROU

NAGAI SEIJI

TANAKA TOSHIHARU

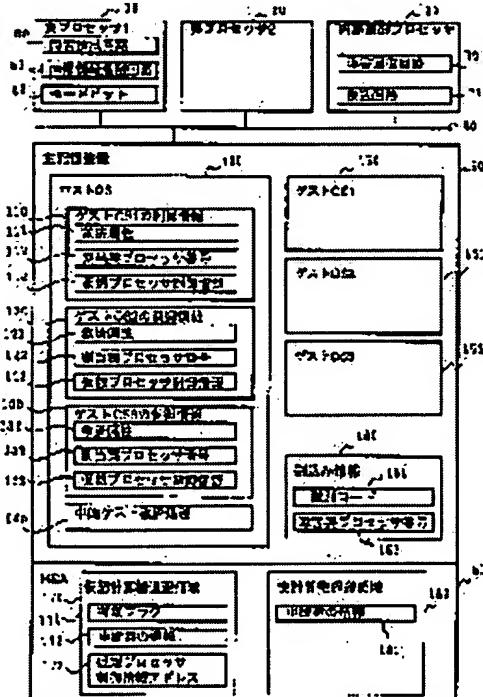
KINOSHITA TOSHIYUKI

(54) FAULT RECOVERY PROCESSING METHOD

(57) Abstract:

PURPOSE: To recover interrupted processing by retrieving a virtual machine, to which a real processor turned to a fixed fault is allocated, selecting and allocating the normal real processor based on the relief attribute of this virtual machine.

CONSTITUTION: Real processors 10 and 20 and an internal control processor 30 are connected through an internal bus 40 to a main storage device 50 and a storage area (HSA) 51 of a hardware and in order to recover the virtual machine interrupted by the fixed faults of the real computers 10 and 20, guest OS relief attributes 111, 121 and 131 expressing the importance of job at the virtual machine are set into control information 110-120 and 130 at respective guest OS 1-3 provided in a host OS storage area 100 inside the main storage device 50. Then, the virtual computer (guest), to which the real processor turned to the fixed fault is allocated, is retrieved and the normal real processor is selected and newly allocated based on the relief attributes applied to this virtual machine so that virtual machine system can be relieved from the fixed fault of the real processor.



BEST AVAILABLE COPY

BEST AVAILABLE COPY

LEGAL STATUS

[Date of request for examination] 06.05.1999

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3196004

[Date of registration] 08.06.2001

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right] 08.06.2005

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平8-263454

(43)公開日 平成8年(1996)10月11日

(51)Int.Cl.⁸ 識別記号 廷内整理番号 F I 標示箇所
G 06 F 15/16 470 G 06 F 15/16 470 R

審査請求 未請求 請求項の数10 FD (全25頁)

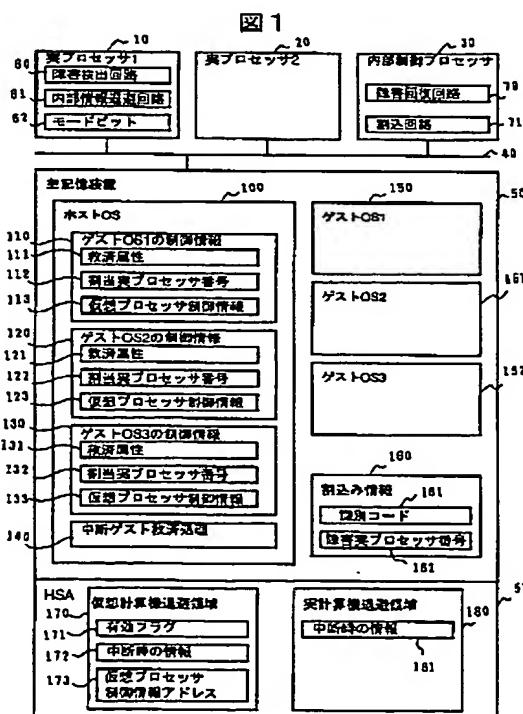
(21)出願番号	特願平7-90314	(71)出願人	000005108 株式会社日立製作所 東京都千代田区神田駿河台四丁目6番地
(22)出願日	平成7年(1995)3月23日	(72)発明者	能田 正 神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内
		(72)発明者	吉岡 正亮郎 神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内
		(72)発明者	長井 清治 神奈川県秦野市堀山下1番地 株式会社日立製作所汎用コンピュータ事業部内
		(74)代理人	弁理士 笹岡 茂 (外1名)
			最終頁に続く

(54)【発明の名称】 障害回復処理方法

(57)【要約】

【目的】 マルチプロセッサ構成の仮想計算機システムで、実プロセッサ固定障害発生時に、仮想計算機の障害時の処理の回復とその業務の救済を可能にする。

【構成】 実プロセッサに固定障害が発生したとき、該実プロセッサが割り当てられた仮想計算機の状態情報および該状態情報の有効・無効を示す有効フラグと、該実プロセッサの状態情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、割込みを受理した正常実プロセッサは、割込み情報に従って実計算機退避領域の状態情報に基づき障害実プロセッサの処理を回復し、かつ前記仮想計算機退避領域の有効フラグが有効を示すとき該退避領域の状態情報に基づき障害仮想計算機の処理を回復し、該回復した仮想計算機の救済属性に基づき新たに割り当てられる正常実プロセッサを選択し、該正常実プロセッサにおける仮想プロセッサの割り当て構成を変更して、仮想計算機システムを救済する。



【特許請求の範囲】

【請求項 1】 マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおける障害回復処理方法において、

実プロセッサに固定障害が発生したとき、該実プロセッサが割り当てられた仮想計算機の状態情報をおよび該状態情報の有効・無効を示す有効フラグと、該実プロセッサの状態情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、

割り込みを受け付けた正常な実プロセッサは、前記主記憶装置に格納された割込み情報と実計算機退避領域の実プロセッサの状態情報に基づき障害実プロセッサの処理を回復し、かつ前記仮想計算機退避領域の有効フラグが有効を示すとき該退避領域の仮想計算機の状態情報に基づき障害仮想計算機の処理を回復し、

固定障害となった実プロセッサが割り当てられている仮想計算機を検索し、該検索した仮想計算機に与えられた救済属性に基づき該仮想計算機に新たに割り当てられる正常な実プロセッサを選択し、

該正常な実プロセッサにおける仮想プロセッサの割り当て構成を変更することにより実プロセッサの固定障害から仮想計算機システムを救済することを特徴とする仮想計算機システムの障害回復処理方法。

【請求項 2】 請求項 1 記載の障害回復処理方法において、

実プロセッサに固定障害が発生したときに仮想計算機が走行している場合は、仮想計算機の状態情報として中断時の情報および仮想計算機制御アドレスと、該状態情報が有効であることを示す有効フラグを前記仮想計算機退避領域に格納し、

実プロセッサが走行している場合は、仮想計算機の状態情報が無効であることを示す有効フラグを前記仮想計算機退避領域に、該実プロセッサの中断時の情報を前記実計算機退避領域に格納することを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項 3】 請求項 2 記載の障害回復処理方法において、

前記障害仮想計算機の処理の回復は、

前記実プロセッサの障害回復完了後、前記仮想計算機退避領域内の有効フラグを検査し、実プロセッサの障害により中断状態にある仮想プロセッサの存在を確認し、

中断状態にある仮想プロセッサが存在する場合、仮想計算機退避領域に格納されている仮想プロセッサの中断時の情報と仮想プロセッサ制御情報アドレスを、主記憶装置へ読み出し、

該読み出した仮想プロセッサの中断時の情報を、仮想計算機退避領域に格納されていた仮想プロセッサ制御情報アドレスへ反映させることを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項 4】 請求項 3 記載の障害回復処理方法において、

正常プロセッサに通知された障害実プロセッサ番号と、仮想計算機の制御プログラムが管理している仮想計算機の制御情報を用い、固定障害となった実プロセッサが割り当てられていた仮想計算機を判断することを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項 5】 請求項 1 記載の障害回復処理方法において、

各実プロセッサ上で走行する各仮想計算機の制御プログラムが管理する仮想計算機の制御情報に仮想計算機の救済属性を設定し、

固定障害となった実プロセッサが割り当てられている仮想計算機の救済属性と、他の正常な実プロセッサを割り当てられている仮想計算機の救済属性とを比較し、該比較の結果に基づき、固定障害となった実プロセッサが割り当てられている実行可能状態に回復された仮想プロセッサへ割り当てる正常な実プロセッサの選択を行うことを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項 6】 請求項 5 記載の障害回復処理方法において、

前記各仮想計算機の救済属性として、重要または非重要を設定し、

前記比較の結果、固定障害となった実プロセッサが割り当てられている仮想計算機の救済属性が重要で、他の正常な実プロセッサを割り当てる正常な実プロセッサを割り当てる正常な実プロセッサに固定障害となった実プロセッサが割り当てられている仮想計算機を停止し、該他の正常な実プロセッサに固定障害となった実プロセッサが割り当てる正常な実プロセッサを割り当てる正常な実プロセッサへ割り当てる正常な実プロセッサの選択を行なうことを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項 7】 請求項 5 記載の障害回復処理方法において、

前記各仮想計算機の救済属性として、固定障害となった実プロセッサが割り当てる正常な実プロセッサを救済した場合における各仮想計算機の正常実プロセッサの使用比を設定し、

固定障害となった実プロセッサが割り当てる正常な実プロセッサを救済しないことを特徴とする仮想計算機システムにおける障害回復処理方法。

サの使用率を割り当て、残りのプロセッサ使用率を前記実行可能状態の各障害仮想計算機にその救済時の実プロセッサ使用比に従い割り当て、該障害仮想計算機を救済し、

割り当てる正常プロセッサがない場合には、前記実行可能状態の障害仮想プロセッサを救済しないことを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項8】 請求項5記載の障害回復処理方法において、

前記各仮想計算機の救済属性として、重要または非重要を設定し、かつ非重要が設定されている場合には固定障害となった実プロセッサが割り当てられている仮想計算機を救済した場合における各仮想計算機の正常実プロセッサの使用比をさらに設定し、

前記比較の結果、固定障害となった実プロセッサが割り当てられている仮想計算機の救済属性が重要で、他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が非重要な場合は、該他の正常な実プロセッサを割り当てられている仮想計算機を停止し、該他の正常な実プロセッサに固定障害となった実プロセッサが割り当てられている仮想計算機を割り当て走行させ、

前記比較の結果、他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が重要な場合は、固定障害となった実プロセッサが割り当てられている仮想計算機の救済を行なわず、

前記比較の結果、固定障害となった実プロセッサが割り当てられている仮想計算機の救済属性および他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が非重要な場合は、固定障害となった実プロセッサが割り当てられている実行可能状態の障害仮想計算機へ、正常な実プロセッサを割り当てるとき、正常動作している仮想計算機に該仮想計算機の救済時の実プロセッサ使用比に従い実プロセッサの使用率を割り当て、残りのプロセッサ使用率を前記実行可能状態の各障害仮想計算機にその救済時の実プロセッサ使用比に従い割り当て、該障害仮想計算機を救済し、

割り当てる正常プロセッサがない場合には、前記実行可能状態の障害仮想プロセッサを救済しないことを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項9】 マルチプロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおける障害回復処理方法において、

実プロセッサに固定障害が発生したとき、該障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の中断時の情報および仮想計算機制御情報アドレスと、該障害実プロセッサの中断時の情報を夫々主記憶装置の仮想計算機逃避領域と実計算機逃避領域に格納し、

障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、

割り込みを受け付けた正常な実プロセッサは、前記割込み情報に基づき障害解析を行ない、障害実プロセッサが割り当てられた論理区画計算機の制御情報に該論理区画計算機のゲストOSのゲスト割込みとしてのゲスト割込みコードを格納し、

前記正常な実プロセッサにおいてホストOS走行モードから前記論理区画計算機のゲストOS走行モードに切り替わったとき、該ゲストOSは該ゲストOSに関連する前記ゲスト割込みコードを検出し、該割込みの発生により前記仮想計算機逃避領域の情報に基づき中断回復処理を行ない、

該中断回復処理後、前記障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の実行を再開することを特徴とする論理区画計算機システムにおける障害回復処理方法。

【請求項10】 マルチプロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおける障害回復処理方法において、

実プロセッサに固定障害が発生したとき、該障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の中断時の情報および仮想計算機制御情報アドレスと、該障害実プロセッサの中断時の情報を夫々主記憶装置の仮想計算機逃避領域と実計算機逃避領域に格納し、

障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、

割り込みを受け付けた正常な実プロセッサは、前記割込み情報に基づき障害解析を行ない、障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の制御情報に該論理区画計算機のゲストOSのゲスト割込みとしてのゲスト割込みコードを格納し、

障害実プロセッサの障害が解消され、障害が解消された実プロセッサにおいてホストOS走行モードから実行待ち状態にある前記論理区画計算機のゲストOS走行モードに切り替わったとき、該ゲストOSは該ゲストOSに関連する前記ゲスト割込みコードを検出し、該割込みの発生により前記仮想計算機逃避領域の情報に基づき中断回復処理を行ない、

該中断回復処理後、前記障害実プロセッサが割り当てられた実行待ち状態にある論理区画計算機内の仮想計算機の実行を再開することを特徴とする論理区画計算機システムにおける障害回復処理方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、計算機システムの障害回復処理方法に係り、特に、仮想計算機システム動作中

に実プロセッサで固定障害が発生したときの障害回復処理方法および論理区画計算機システム動作中に実プロセッサで固定障害が発生したときの障害回復処理方法に関するものである。

【0002】

【従来の技術】1台の実計算機上で複数の仮想的な計算機（以下では、仮想計算機またはゲストとも呼ぶ）を同時に走行させるシステムとして、仮想計算機システムがある。仮想計算機システムでは、ゲストを制御する制御プログラムであるホストOS（以下では、実計算機のハードウェアとホストOSと合わせてホストとも呼ぶ）が走行し、ゲストのスケジューリング、ディスパッチ、命令シミュレーションなどを制御している。また、仮想計算機システムにおけるホストOSをハードウェア機構のように提供し、実計算機を論理的に分割しているようにユーザから見える論理分割システムがある。仮想計算機走行中の状態をゲストモードと呼び、ホストOS走行中の状態をホストモードと呼ぶ。実プロセッサが、ホストモードからゲストモードへ切り替わると、ホストモード実行に関する実行情報を、実プロセッサから、主記憶装置に含まれるプログラム記憶域（PSA）またはハードウェアの記憶領域（HSA）へ格納し、ゲストモード実行に関する実行情報を主記憶から実プロセッサへ読み込む。また、実プロセッサが、ゲストモードからホストモードへ切り替わると、実プロセッサから、ゲストモード実行に関する実行情報を主記憶へ格納し、ホストモード実行に関する実行情報を、主記憶装置に含まれるPSAまたはHSAから実プロセッサへ読み込む。

【0003】これら仮想計算機システム、論理分割システムに関しては、特開昭57-212680号公報、特開昭64-37636号公報、特開平2-96246号公報にその一例が記載されている。仮想計算機システムにおいて、ホストOSの障害、本体系障害により中断した仮想計算機を救済する、仮想計算機システムの障害救済技術がある。例えば、仮想計算機システムの本体系障害発生時の処理方式に関して、特開昭64-53238号公報がある。複数の仮想計算機を制御するVMモニタ（ホストOS）側に、本体系障害に遭遇した仮想計算機のプロセスの回復処理を行なう機能を持たせ、仮想計算機のプロセス回復処理機能を持つ必要をなくしている。また、仮想計算機システムの回復制御方式に関して、特開昭64-3741号公報にその一例がある。実行権のない待機中のゲスト情報をメモリ中に退避し、仮想計算機システムにおけるシステムダウン発生時、待機中のゲストを再立ち上げすることなく、環境の回復を行なう。

【0004】従来のマルチプロセッサ構成の計算機で動作する仮想計算機システムにおける、実プロセッサの固定障害に対する障害回復技術について、図2を用い説明する。図2において、200～201は実プロセッサ、202は計算機内部動作を制御する内部制御プロセッ

サ、204は前記プロセッサが共有する主記憶装置、205は前記プロセッサのハードウェアの記憶領域（HSA）、210はプロセッサ内の障害検出回路、220は内部制御プロセッサ内の障害情報格納回路、230は障害発生通知回路である。203はプロセッサ200～201と内部制御プロセッサ202と主記憶装置204とHSA205を接続する内部バスである。240は仮想計算機システムを制御するホストOSの主記憶領域、250は仮想計算機動作を制御するゲストOS1の制御情報、251はゲストOS1の仮想プロセッサ制御情報、260はゲストOS2の制御情報、261はゲストOS2の仮想プロセッサ制御情報、270は仮想計算機障害回復処理である。280は主記憶装置204内のゲストOS1の主記憶領域、281は主記憶装置204内のゲストOS2の主記憶領域、290はHSA205内の中断処理の障害情報格納領域である。

【0005】仮想計算機システムの実プロセッサで固定障害が発生したとき、走行していた仮想計算機を救済する方法の一例を示す。ゲストOS1とゲストOS2はシングルプロセッサ構成の仮想計算機であり、それぞれの仮想プロセッサはプロセッサ1とプロセッサ2を割当てられている。実プロセッサ1がゲストOS1の仮想プロセッサとして動作していたとき、実プロセッサ1上で固定障害が発生したとすると、実プロセッサ1内の障害検出回路210が障害を検出し、内部制御プロセッサ202に報告する。内部制御プロセッサ202の障害情報格納回路220は、実プロセッサ1が保持している命令アドレス、各レジスタの内容を中断処理の障害情報格納領域290へ退避する。退避完了後、障害発生通知回路230が障害発生をホストOSへ通知する。中断情報からホストOSが中断処理を回復する。

【0006】また、区画区分式プロセス環境における区画間制御のための方法および装置に関しては、特開平5-181823号公報にその一例がある。論理区画ごとにシステムを確立し、各論理区画のシステム動作を別の論理区画のシステムに監視させ、あるシステムが故障したときそのシステムの故障の際の応答動作を許可されているシステムが、故障システムのシステムリセット、非活動化等の処理を行なわせる。

【0007】次に、従来のマルチプロセッサ構成の計算機で動作する論理区画計算機システムにおける、実プロセッサの固定障害に対する障害回復技術について、図15を用い説明する。なお、従来のマルチプロセッサ構成の計算機で動作する仮想計算機システムでは、マルチプロセッサとして実プロセッサ1と実プロセッサ2があり、ゲストOSが仮想プロセッサ1と仮想プロセッサ2からなるマルチプロセッサ構成である場合に、一方の実プロセッサにゲストOSの仮想プロセッサ1及び2を割当ることも、仮想プロセッサ1を一方の実プロセッサに割り当て、仮想プロセッサ2を他方の実プロセッサに

割り当てもできる。しかし、従来のマルチプロセッサ構成の計算機で動作する論理区画計算機システムでは、仮想プロセッサ1を一方の実プロセッサに割り当て、仮想プロセッサ2を他方の実プロセッサに割り当てることはできるが、一方の実プロセッサにゲストOSの仮想プロセッサ1及び2を割り当てることはできない。

【0008】図15において、1200～1201は実プロセッサ、1202は計算機内部動作を制御する内部制御プロセッサ、1204は前記プロセッサが共有する主記憶装置、1205は前記プロセッサのハードウェアの記憶領域(HSA)、1210は実プロセッサ内の障害検出回路、1220は実プロセッサ内の障害情報格納回路、1230は障害発生通知回路である。1203はプロセッサ1200～1201と内部制御プロセッサ1202と主記憶装置1204とHSA1205を接続する内部バスである。1240は論理区画計算機システムを制御するホストOSの主記憶領域、1250は論理区画計算機動作を制御するゲストOS1の制御情報、1251はゲストOS1の仮想プロセッサ1制御情報、1252はゲストOS1の仮想プロセッサ2制御情報、1260はゲストOS2の制御情報、1261はゲストOS2の仮想プロセッサ1制御情報、1265はゲストOS3の制御情報、1266はゲストOS3の仮想プロセッサ1制御情報、1270は実プロセッサ割り当てテーブル、1275は論理区画計算機救済処理である。1280は主記憶装置1204内のゲストOS1の主記憶領域、1281はゲストOS2の主記憶領域、1282はゲストOS3の主記憶領域、1290はHSA1205内の中断処理の情報格納領域である。

【0009】論理区画計算機システムの実プロセッサで固定障害が発生したとき、走行していた論理区画計算機を救済する方法の一例を示す。ゲストOS1はマルチプロセッサ構成の論理区画計算機、ゲストOS2とゲストOS3はシングルプロセッサ構成の論理区画計算機である。ゲストOS1において、ゲストOS1の仮想プロセッサ1は実プロセッサ1、ゲストOS1の仮想プロセッサ2は実プロセッサ2に割り当てられている。ゲストOS2の仮想プロセッサ1は実プロセッサ1に割り当てられ、ゲストOS3の仮想プロセッサ1は実プロセッサ2に割り当てられている。実プロセッサ1がゲストOS1の仮想プロセッサ1として動作していたとき、実プロセッサ1上で固定障害が発生したとすると、実プロセッサ1内の障害検出回路1210が障害を検出し、内部制御プロセッサ1202に報告する。内部情報退避回路1220は、実プロセッサ1が保持している命令アドレス、各レジスタの内容を中断処理の情報格納領域1290へ退避する。退避完了後、障害通知回路1230が障害発生をホストOSへ通知する。中断処理の情報格納領域1290により、ホストOSの論理区画計算機救済処理が、中断処理を回復する。

【0010】

【発明が解決しようとする課題】従来の单一OSが走行するマルチプロセッサシステムのプロセッサで固定障害が発生した場合、正常プロセッサが、障害の発生したプロセッサで実行していた処理を、一時的に処理を引き継ぐことによりシステムの停止を救済していた。一時的に救済すればシステムダウンを回避できるが、プロセッサの固定障害が発生した時の正常プロセッサによる処理の引継ぎは、障害に対するプロセッサ構成の縮退制御である。このため、正常プロセッサで処理を引き継いでいる間、システムの性能が非常に低下する。このため仮想計算機システムに適用する場合、ゲストOSの仮想プロセッサ構成と性能への影響が問題となる。

【0011】しかし、従来の仮想計算機システムにおける本体系障害発生時の処理方式においても、障害仮想プロセッサと正常仮想プロセッサが実プロセッサを共有し、障害仮想計算機の中断を回復するため、プロセッサ台数の減少により実計算機システムの性能が低下し、ゲスト性能を低下させてしまう。性能を保証する必要がある仮想計算機を救済するとき、救済する仮想計算機の性能を保証する必要があるとき、性能の確保が問題となる。

【0012】従来の单一OSが走行するマルチプロセッサ構成の計算機システムにおける、プロセッサの固定障害による中断処理の継続方式では、同一OSが走行する正常プロセッサにより中断処理を回復するので、障害となったプロセッサ内で中断した処理の情報だけを退避していた。しかし、仮想計算機システムへ適用する場合には、障害により中断した処理がホストOSかゲストOSかの判断と、影響を受ける停止状態ゲストの判断が必要となる。従来の仮想計算機システムにおける本体系障害発生時の処理方式では、ホストモードの実プロセッサが固定障害となった場合、ホストモード実行に関する実行情報を退避することが可能である。しかし、その実プロセッサで実行されるのを待っている実行可能状態の仮想プロセッサなど識別する手段がない。また、従来の仮想計算機システムにおける本体系障害発生時の処理方式では、ゲストモードの実プロセッサが固定障害となった場合、仮想プロセッサ実行に関する実行情報を退避することが可能であるが、主記憶装置に含まれるプログラム記憶域(PSA)またはハードウェアの記憶領域(HSA)へ格納されているホストOSの実行内容を保証する手段がない。

【0013】また、従来の仮想計算機システムにおける本体系障害発生時の処理方式では、仮想プロセッサが走行している実プロセッサが固定障害となった場合、中断情報を退避し、ホストOSが障害仮想計算機のOSへ障害を通知し、ゲストOSが中断処理を回復する。または、ホストOSが、走行中に障害となったゲストOSの中断処理を回復する。しかし、複数の種類のゲストOS

の中断処理を救済する場合、各ゲストOSの種類やバージョンにあった救済が必要であり、ホストOSの対応が困難である。また、走行中障害となった処理の回復方法は提案されているが、障害となった実プロセッサを割当られている仮想プロセッサ上で、実行待ち状態であった処理を救済する手段がない。

【0014】本発明の目的は、マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおいて、仮想プロセッサとして動作していた実プロセッサが固定障害となったとき、中断した仮想計算機計算機の処理を回復する方法を提供することにある。本発明の他の目的は、マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおいて、仮想プロセッサとして動作中の実プロセッサが固定障害となったとき、動作中であった仮想プロセッサの状態を保証する方法を提供することにある。本発明の他の目的は、マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおいて実プロセッサが固定障害となったとき、中断した仮想計算機計算機の処理を回復し、中断した仮想プロセッサを実行可能状態へ移行する方法を提供することにある。本発明の目的は、マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおいて、実プロセッサに障害が発生したとき、重要な仮想計算機の業務を救済する方法を提供することにある。本発明の他の目的は、マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおいて、実プロセッサに障害が発生したとき、仮想計算機が要求する処理性能を保証する方法を提供することにある。本発明の他の目的は、マルチプロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおいて、仮想プロセッサとして動作していた実プロセッサが固定障害となったとき、動作中であった論理区画計算機を回復する方法を提供することにある。また、本発明の他の目的は、マルチプロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおいて、実プロセッサが固定障害となったとき、障害となった実プロセッサを割当られている仮想プロセッサ上で、実行待ち状態であった処理を救済する方法を提供することにある。

【0015】

【課題を解決するための手段】上記目的を達成するため、本発明は、マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおける障害回復処理方法において、実プロセッサに固定障害が発生したとき、該実プロセッサが割り当られた仮想計算機の状態情報および該状態情報の有効・無効を示す有効フラグと、該実プロセッサの状態情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、障害の発生を正常な実プロセッサに報告し、該正常な実

プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、割り込みを受け付けた正常な実プロセッサは、前記主記憶装置に格納された割込み情報と実計算機退避領域の実プロセッサの状態情報に基づき障害実プロセッサの処理を回復し、かつ前記仮想計算機退避領域の有効フラグが有効を示すとき該退避領域の仮想計算機の状態情報に基づき障害仮想計算機の処理を回復し、固定障害となった実プロセッサが割り当てられている仮想計算機を検索し、該検索した仮想計算機に与えられた救済属性に基づき該仮想計算機に新たに割り当てられる正常な実プロセッサを選択し、該正常な実プロセッサにおける仮想プロセッサの割り当て構成を変更することにより実プロセッサの固定障害から仮想計算機システムを救済するようとしている。さらに、実プロセッサに固定障害が発生したときに仮想計算機が走行している場合は、仮想計算機の状態情報として中断時の情報および仮想計算機制御アドレスと、該状態情報が有効であることを示す有効フラグを前記仮想計算機退避領域に格納し、実プロセッサが走行している場合は、仮想計算機の状態情報が無効であることを示す有効フラグを前記仮想計算機退避領域に、該実プロセッサの中断時の情報を前記実計算機退避領域に格納するようとしている。さらに、前記障害仮想計算機の処理の回復は、前記実プロセッサの障害回復完了後、前記仮想計算機退避領域内の有効フラグを検査し、実プロセッサの障害により中断状態にある仮想プロセッサの存在を確認し、中断状態にある仮想プロセッサが存在する場合、仮想計算機退避領域に格納されている仮想プロセッサの中断時の情報と仮想プロセッサ制御情報アドレスを、主記憶装置へ読み出し、該読み出した仮想プロセッサの中断時の情報を、仮想計算機退避領域に格納されていた仮想プロセッサ制御情報アドレスへ反映させるようとしている。さらに、正常プロセッサに通知された障害実プロセッサ番号と、仮想計算機の制御プログラムが管理している仮想計算機の制御情報を用い、固定障害となった実プロセッサが割り当っていた仮想計算機を判断するようとしている。さらに、各実プロセッサ上で走行する各仮想計算機の制御プログラムが管理する仮想計算機の制御情報を仮想計算機の救済属性を設定し、固定障害となった実プロセッサが割り当られている仮想計算機の救済属性と、他の正常な実プロセッサを割り当られている仮想計算機の救済属性とを比較し、該比較の結果に基づき、固定障害となった実プロセッサが割り当られている実行可能状態に回復された仮想プロセッサへ割り当てる正常な実プロセッサの選択を行なうようとしている。さらに、前記各仮想計算機の救済属性として、重要または非重要を設定し、前記比較の結果、固定障害となった実プロセッサが割り当られている仮想計算機の救

済属性が重要で、他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が非重要な場合は、該他の正常な実プロセッサを割り当てられている仮想計算機を停止し、該他の正常な実プロセッサに固定障害となった実プロセッサが割り当てられている仮想計算機を割り当て走行させ、前記比較の結果、他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が重要な場合は、固定障害となった実プロセッサが割り当てられている仮想計算機を救済しないようにしている。また、前記各仮想計算機の救済属性として、固定障害となった実プロセッサが割り当てられている仮想計算機を救済した場合における各仮想計算機の正常実プロセッサの使用比を設定し、固定障害となった実プロセッサが割り当てられている実行可能状態の障害仮想計算機へ、正常な実プロセッサを割り当てるとき、正常動作している仮想計算機に該仮想計算機の救済時の実プロセッサ使用比に従い実プロセッサの使用率を割り当て、残りのプロセッサ使用率を前記実行可能状態の各障害仮想計算機にその救済時の実プロセッサ使用比に従い割り当て、該障害仮想計算機を救済し、割り当てる正常プロセッサがない場合には、前記実行可能状態の障害仮想プロセッサを救済しないようにしている。

【0016】また、前記各仮想計算機の救済属性として、重要または非重要を設定し、かつ非重要が設定されている場合には固定障害となった実プロセッサが割り当てられている仮想計算機を救済した場合における各仮想計算機の正常実プロセッサの使用比をさらに設定し、前記比較の結果、固定障害となった実プロセッサが割り当てられている仮想計算機の救済属性が重要で、他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が非重要な場合は、該他の正常な実プロセッサを割り当てられている仮想計算機を停止し、該他の正常な実プロセッサに固定障害となった実プロセッサが割り当てられている仮想計算機を割り当て走行させ、前記比較の結果、他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が重要な場合は、固定障害となった実プロセッサが割り当てられている仮想計算機の救済を行なわず、前記比較の結果、固定障害となった実プロセッサが割り当てられている仮想計算機の救済属性および他の正常な実プロセッサを割り当てられている仮想計算機の救済属性が非重要な場合は、固定障害となった実プロセッサが割り当てられている実行可能状態の障害仮想計算機へ、正常な実プロセッサを割り当てるとき、正常動作している仮想計算機に該仮想計算機の救済時の実プロセッサ使用比に従い実プロセッサの使用率を割り当て、残りのプロセッサ使用率を前記実行可能状態の各障害仮想計算機にその救済時の実プロセッサ使用比に従い割り当て、該障害仮想計算機を救済し、割り当てる正常プロセッサがない場合には、前記実行可能状態の障害仮想プロセッサを救済しないようにしている。また、マルチ

プロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおける障害回復処理方法において、実プロセッサに固定障害が発生したとき、該障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の中断時の情報および仮想計算機制御情報アドレスと、該障害実プロセッサの中断時の情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、割り込みを受け付けた正常な実プロセッサは、前記割込み情報に基づき障害解析を行ない、障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の制御情報に該論理区画計算機のゲストOSのゲスト割込みとしてのゲスト割込みコードを格納し、前記正常な実プロセッサにおいてホストOS走行モードから前記論理区画計算機のゲストOS走行モードに切り替わったとき、該ゲストOSは該ゲストOSに関連する前記ゲスト割込みコードを検出し、該割込みの発生により前記仮想計算機退避領域の情報に基づき中断回復処理を行ない、該中断回復処理後、前記障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の実行を再開するようにしている。また、マルチプロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおける障害回復処理方法において、実プロセッサに固定障害が発生したとき、該障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の中断時の情報および仮想計算機制御情報アドレスと、該障害実プロセッサの中断時の情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、割り込みを受け付けた正常な実プロセッサは、前記割込み情報に基づき障害解析を行ない、障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の制御情報に該論理区画計算機のゲストOSのゲスト割込みとしてのゲスト割込みコードを格納し、障害実プロセッサの障害が解消され、障害が解消された実プロセッサにおいてホストOS走行モードから実行待ち状態にある前記論理区画計算機のゲストOS走行モードに切り替わったとき、該ゲストOSは該ゲストOSに関連する前記ゲスト割込みコードを検出し、該割込みの発生により前記仮想計算機退避領域の情報に基づき中断回復処理を行ない、該中断回復処理後、前記障害実プロセッサが割り当てられた実行待ち状態にある論理区画計算機内の仮想計算機の実行を再開するようにしている。

【0017】

【作用】上記手段により、マルチプロセッサ構成上で稼動する仮想計算機システムにおいて、実プロセッサで固定障害が発生したときの実プロセッサおよび仮想計算機の障害回復が可能となる。また、重要度の高い仮想計算機を処理性能を低下させることなく救済することが可能となる。また、同じ重要度の複数の仮想計算機を救済時の実プロセッサ使用比にしたがってプロセッサ構成を変更することなく救済することが可能となる。さらに、マルチプロセッサ構成上で複数の論理区画計算機が走行する論理区画計算機システムにおいて、固定障害となった実プロセッサが割り当てられた論理区画計算機の仮想プロセッサを論理区画計算機のゲストOSにより中断状態から回復することが可能となる。また、実プロセッサが交換、リセット等により障害解消されたときに、実プロセッサが障害のために実行待ち状態であった論理区画計算機の仮想プロセッサの処理を救済することが可能となる。

【0018】

【実施例】図面を用い本発明の実施例を示し、詳細に説明する。第1の実施例について、図1、図3～図10を参照して説明する。図1は実施例の構成図であり、1台のマルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおいて、実プロセッサの固定障害により中断した仮想計算機を回復する仮想計算機システムの構成図である。

【0019】図1において、10は実プロセッサ1、20は実プロセッサ2、30は内部制御プロセッサ、50は主記憶装置、51はハードウェアの記憶領域(HSA)、40は前記プロセッサと内部制御プロセッサと主記憶装置とHSAを接続する内部バスである。実施例では、内部制御プロセッサ30を設けているが、内部制御プロセッサ30を設げずに、該プロセッサの機能を実プロセッサ1及び実プロセッサ2に持たせるようにしてもよい。60は実プロセッサ1に組み込まれる障害検出回路、61は実プロセッサ1に組み込まれる内部情報退避回路、62は実プロセッサにおける実プロセッサが走行している実計算機モードと仮想計算機が走行している仮想計算機モードを示す実プロセッサ1に組み込まれるモードビットである。本実施例では、障害検出回路60および内部情報退避回路61およびモードビット62を、実プロセッサ1の内部にのみ示しているが、実プロセッサ2にも組み込まれるものである。70は内部制御プロセッサに組み込まれる障害回復回路、71は前記実プロセッサへ割込み信号を発生させる割回路である。100は主記憶装置50内のゲストOS記憶領域、110はゲストOS1の制御情報、120はゲストOS2の制御情報、130はゲストOS3の制御情報である。夫々のゲストOSの制御情報内に、111、121、131の救済属性、112、122、132の割当実プロセッ

サ番号、113、123、133の仮想プロセッサ制御情報が含まれる。140はホストOS内の中断ゲスト救済処理である。150～152は主記憶装置50内のゲストOS1～OS3の主記憶領域、160は割込み情報領域、161は割込みの内容(例えば、プロセッサ、メモリ、I/O等)を示す識別コード格納領域、162は障害実プロセッサ番号格納領域である。170はHSA51内に確保される障害発生時の仮想計算機の状態情報を格納する仮想計算機退避領域であり、171は仮想計算機退避領域の有効状態を示す有効フラグ、172は仮想計算機退避領域内の仮想プロセッサ中断時の情報を格納する領域、173は仮想計算機退避領域内の中断した仮想プロセッサ制御情報アドレスを格納する領域である。180はHSA51内に確保される障害発生時の実プロセッサの状態情報を格納する実計算機退避領域であり、181は実計算機退避領域内の実プロセッサ中断時の情報を格納する領域である。

【0020】本発明では、実プロセッサの固定障害により中断した仮想計算機を回復するため、仮想計算機の業務の重要性を表すゲストOS救済属性111、121、131を設定するところに特徴がある。また、実プロセッサの固定障害が、モードビット62を用い実計算機走行中か仮想計算機走行中であるか識別し、仮想計算機退避領域170および実計算機退避領域180へ実計算機状態と仮想計算機状態を退避し保証するするところに特徴がある。

【0021】図3は、実プロセッサに固定障害が発生した時に動作していた仮想プロセッサの状態を退避する仮想計算機退避領域170である。仮想プロセッサ中断時の情報172内に、プログラム状態語310、汎用レジスタの内容311、浮動小数点レジスタの内容312、制御レジスタの内容313、タイマ値314などが退避される。図4は、実プロセッサに固定障害が発生した時に動作していた実計算機の状態を退避する実計算機退避領域180である。実プロセッサ中断時の情報181内に、プログラム状態語320、汎用レジスタの内容321、浮動小数点レジスタの内容322、制御レジスタの内容323、タイマ値324などが退避される。図5は、実施例の仮想計算機の構成と、各ゲストOSの制御情報内に設定されている救済属性の例と、ゲストOSに割当られている実プロセッサ番号の例を表330に示したものである。である。ゲストOS1が重要ゲスト属性、ゲストOS2とゲストOS3が通常ゲスト属性である。ゲストOS1に実プロセッサ1、ゲストOS2とゲストOS3に実プロセッサ2が割当られている。

【0022】次に、実プロセッサ1において、固定障害が発生した時の仮想計算機システム回復方法を、以下説明する。実プロセッサ1の内部情報退避回路61における処理を、図6を用いて説明する。従来技術で提供されている障害検出回路60により、実プロセッサ1で発生し

た固定障害が検出される（ステップ400）。中断した命令の実行内容を保証するため、実プロセッサ1の命令実行マイクロプログラムを停止する（ステップ401）。内部制御プロセッサに固定障害の検出を報告する（ステップ402）。内部情報を退避する前に、モードビット62により実プロセッサ上で仮想計算機が走行中か検査する（ステップ403）。実プロセッサ1上で仮想計算機が走行していた場合、仮想プロセッサの中断情報を仮想計算機退避領域170へ退避する（ステップ404）。中断した仮想プロセッサの制御情報アドレスを仮想計算機退避領域170へ退避する（ステップ405）。中断時の情報172および仮想プロセッサ制御情報アドレス173への退避完了後、仮想計算機退避領域170の有効フラグ171を設定する（ステップ406）。仮想計算機の状態退避完了後、実計算機の実行情報を退避するため、実計算機モードへ切替える（ステップ407）。ステップ403でモードビット62を検査し、実プロセッサ1上で実計算機が走行、すなわち、ホストOSの一部である仮想計算機制御プログラムが走行、していた場合、仮想計算機退避領域内の有効フラグ領域171へ、仮想計算機退避領域の無効フラグを設定する（ステップ408）。ステップ407またはステップ408の後、実プロセッサの中断時の情報を実計算機退避領域180へ退避する（ステップ409）。そしてチェックストップ状態とする（ステップ410）。

【0023】実プロセッサ1の固定障害を通知される、内部制御プロセッサ30の障害回復回路70における処理を、図7を用いて説明する。障害プロセッサにより固定障害の報告が、障害回復回路70に対し出される（ステップ430）。障害解析を行なう（ステップ431）。障害プロセッサによる内部情報の退避が完了したかどうかをチェックする（ステップ432）。

【0024】障害実プロセッサによる内部情報退避が失敗した場合、以下のステップ434～440を正常の実プロセッサにより処理する。モードビット62により実プロセッサ上で仮想計算機が走行中か検査する（ステップ434）。実プロセッサ1上で仮想計算機が走行していた場合、仮想プロセッサの中断情報を仮想計算機退避領域170へ退避する（ステップ435）。中断した仮想プロセッサの制御情報アドレスを仮想計算機退避領域170へ退避する（ステップ436）。中断時の情報172および仮想プロセッサ制御情報アドレス173への退避完了後、仮想計算機退避領域170の有効フラグ171を設定する（ステップ437）。仮想計算機の状態退避完了後、実計算機の実行情報を退避するため、実計算機モードへ切替える（ステップ438）。ステップ403でモードビット62を検査し、実プロセッサ1上で実計算機が走行していた場合、仮想計算機退避領域内の有効フラグ領域171へ、仮想計算機退避領域の無効フラグを設定する（ステップ439）。ステップ407ま

たはステップ408の後、実プロセッサの中断時の情報を実計算機退避領域180へ退避する（ステップ440）。

【0025】ステップ432で成功したと判断したとき、または、ステップ440が完了したとき、障害プロセッサがチェックストップ状態かどうかをチェックする（ステップ441）。もし、チェックストップ状態でない場合、障害実プロセッサをチェックストップにする（ステップ442）。障害実プロセッサがチェックストップ状態であると、プロセッサ障害の割り込み情報160へ障害識別コードと障害実プロセッサ番号を格納する（ステップ443）。その後、従来技術である割込み回路71を使用し、正常プロセッサ上で動作するホストOSへのプロセッサ障害の割り込みを有効にする（ステップ444）。前記割込みにより、ホストOS内の中断ゲスト救済処理140が動作し、中断ゲストを回復、すなわち、中断した状態を復元する。

【0026】プロセッサ障害の割り込みにより起動される、ホストOS内の中断ゲスト救済処理140について、図8～図9を用いて説明する。ホストOS内の中断ゲスト救済処理140は、プロセッサ障害の割り込みを検出すると、正常プロセッサ上で走行中しているゲストOSを停止し、ホストOS内の仮想計算機制御プログラムを動作させる（ステップ460）。その後、障害割込みを受ける（ステップ461）。実計算機退避領域180内の中断時の情報181を、アクセス可能な主記憶へ読み込む（ステップ462）。実計算機の中断時の情報181により、実計算機動作を回復する（ステップ463）。仮想計算機退避領域170の有効フラグ171を検査する（ステップ464）。有効フラグ171が有効であると判断したとき、障害仮想プロセッサの制御情報アドレス173を、ホストOSがアクセス可能な主記憶領域へ読み込む（ステップ465）。そして、仮想計算機退避領域170内の中断時の情報172を、アクセス可能な主記憶領域へ読み込む（ステップ466）。退避されている中断時の情報より仮想プロセッサの中断処理が回復可能であるか判断する（ステップ467）。回復可能であると判断した場合、中断時の情報による中断処理の回復処理を行なう（ステップ468）。回復した中断処理を、障害仮想プロセッサの制御情報アドレスの内容に反映させる（ステップ469）。すなわち、内容の更新をする。ステップ467の判断で回復不可能であると判断した場合、障害仮想プロセッサの障害としてホストOS、ゲストOSに報告する（ステップ470）。ステップ464のチェックで有効フラグが無効であると判断したとき、または、ステップ469が完了したとき、または、ステップ470が完了したとき、割込み情報160の障害実プロセッサ番号162から障害ゲストを検索する（ステップ471）。重要ゲスト属性の障害ゲストOSがあるかチェックする（ステップ472）。重要

ゲスト属性の障害ゲストOSがある場合、障害となった実プロセッサを割当てられている仮想プロセッサを検索する（ステップ473）。重要ゲスト属性の障害ゲストOSがない場合、処理を終了する。重要ゲスト属性のゲストが割当てられていない正常な実プロセッサを検索する（ステップ474）。障害仮想プロセッサに割当てることが可能な正常な実プロセッサがあるかチェックする（ステップ475）。障害仮想プロセッサに割当てることが可能な正常な実プロセッサがある場合、その正常な実プロセッサ上で走行中のゲストがあれば停止する（ステップ476）。障害仮想プロセッサに割当てる正常な実プロセッサがない場合、回復処理を停止する。ステップ476完了後、重要ゲスト属性である障害ゲストの仮想プロセッサに正常な実プロセッサを割当てる（ステップ477）。

【0027】図10は、前記制御により救済された後の仮想計算機システムの構成と、各ゲストOSの制御情報内に設定されている救済属性と、ゲストOSに割当てられている実プロセッサ番号を示した表480である。ゲストOS1は、重要ゲスト属性であるため、実プロセッサ2により救済され継続走行する。ゲストOS2とゲストOS3は、通常ゲスト属性であるため、停止状態となる。並行保守、リセットなどの技術により、実プロセッサの固定障害が取り除かれた後、停止状態のゲストOS2とゲストOS3に、実プロセッサを割当ることにより、停止状態から動作状態とすることが可能である。この場合、ゲストOS2とゲストOS3は実プロセッサの停止状態を意識しないので、動作を継続しているようにOSから見える。

【0028】以上述べたように、本実例によれば、マルチプロセッサ構成の計算機上で稼動する仮想計算機システムにおいて、実プロセッサ上で固定障害が発生しても、性能を確保する必要のある仮想計算機の処理性能を低下させることなく動作を継続させ、システムダウンしないで仮想計算機システムの障害を回復する方法を提供できる。

【0029】次に、第2の実施例について、図11～図13を参照して説明する。

【0030】図11は、実施例の仮想計算機の構成と、各ゲストOSの制御情報内に設定されている救済属性と、ゲストOSに割当てられている実プロセッサ番号と、実プロセッサ使用比を示す表500である。救済属性は、仮想計算機の性能を確保する必要があることを識別する重要属性を含む。また、救済属性は、そのゲストOSが割当てられている正常実プロセッサを障害仮想計算機に提供するとき、そのゲストOSが確保する正常実プロセッサの使用比を設定するための、救済時の実プロセッサ使用比を含む。ゲストOS1、ゲストOS2、ゲストOS3は通常ゲスト属性である。ゲストOS1とゲストOS2に実プロセッサ1、ゲストOS3に実プロセ

ッサ2が割当てられている。ゲストOS1は実プロセッサ使用比90%で動作し、ゲストOS2は実プロセッサ使用比10%で動作し、ゲストOS3は実プロセッサ使用比100%で動作している。ゲストOS1の救済時の実プロセッサ使用比は90%、ゲストOS2の救済時の実プロセッサ使用比は0%、ゲストOS3の救済時の実プロセッサ使用比は20%に設定されている。実プロセッサ1において固定障害が発生し、前記実施例により仮想計算機システムの障害を回復する。しかし、全ゲストOSが通常ゲスト属性であるので、実プロセッサ1を割当てられているゲストOS1、ゲストOS2は停止状態となる。そして、各ゲストOSに定義している救済時の実プロセッサ使用比に従い、停止状態のゲストOS1、ゲストOS2に、実プロセッサを割当て動作を再開する。

【0031】図12は、ゲストOSに定義している救済時の実プロセッサ使用比に従い、障害ゲストOSに、適切な実プロセッサ使用比を割当てる処理であり、中断ゲスト救済処理に含まれ、重要、非重要を救済属性としている図9に示す処理に代わる処理となる。

【0032】障害となった実プロセッサを割当てられている仮想プロセッサを検索する（ステップ510）。重要ゲスト属性のゲストが割当てられていない正常な実プロセッサを検索する（ステップ511）。全障害ゲストに割当てる実プロセッサ使用比を障害ゲストの救済実プロセッサ使用比として、100から正常ゲストOSの救済時の実プロセッサ使用比を減算し、障害ゲストの救済実プロセッサ使用比を計算する（ステップ512）。各障害ゲストに割当てる実プロセッサ使用比を障害回復実プロセッサ使用比として、障害ゲストの救済実プロセッサ使用比に正常時の実プロセッサ使用比を乗じたものを100で除算し、各障害ゲストの障害回復実プロセッサ使用比を計算する（ステップ513）。障害ゲストOSに障害回復実プロセッサ使用比を設定する（ステップ514）。停止中の正常ゲストOSに救済時の実プロセッサ使用比を設定する（ステップ515）。

【0033】図13は、前記制御により救済された後の仮想計算機システムの構成である。仮想計算機の構成と、各ゲストOSの制御情報内に設定されている救済属性と、ゲストOSに割当てられている実プロセッサ番号と、実プロセッサ使用比を示した表530である。救済後、ゲストOS1、ゲストOS2、ゲストOS3に実プロセッサ2が割当てられる。ゲストOS3は、救済時の実プロセッサ使用比と同じ、実プロセッサ使用比20%で動作する。正常状態のゲストOS1、ゲストOS2の実プロセッサ使用比は90%、10%であったので、障害ゲストの救済実プロセッサ使用比の80%を分配し、ゲストOS1は実プロセッサ使用比72%で動作し、ゲストOS2は実プロセッサ使用比8%で動作する。全ゲストOSは実プロセッサの停止状態を意識しないので、

動作を継続しているようにOSから見える。

【0034】なお、救済属性として重要、非重要を用い、非重要が設定されている場合に、さらに、固定障害となった実プロセッサが割り当てられている仮想計算機を救済した場合における各仮想計算機の正常実プロセッサの使用比を救済属性として設定し、固定障害となった実プロセッサが割り当てられている仮想計算機の救済属性および正常実プロセッサの救済属性が共に非重要な場合に、前述の図12の処理をするようにしてもよい。

【0035】以上述べたように、本実例によれば、マルチプロセッサ構成の計算機上で稼動する仮想計算機システムにおいて、実プロセッサ上で固定障害が発生しても、仮想計算機のプロセッサ構成を変更することなく動作を継続させ、システムダウンしないで仮想計算機システムの障害を回復する方法を提供できる。

【0036】次に、第3の実施例について説明する。

【0037】図14は第3の実施例の構成図であり、マルチプロセッサ構成の計算機上で複数の論理区画計算機が走行する論理区画計算機システムにおいて、実プロセッサの固定障害により中断した論理区画計算機を回復する論理区画計算機システムの構成図である。

【0038】図14において、1010は実プロセッサ1、1020は実プロセッサ2、1030は内部制御プロセッサ、1050は主記憶装置、1051はハードウェアの記憶領域(HSA)、1040は前記プロセッサと内部制御プロセッサと主記憶装置とHSAを接続する内部バスである。実施例では、内部制御プロセッサ1030を設けているが、内部制御プロセッサ1030を設けずに、該プロセッサの機能を実プロセッサ1及び実プロセッサ2に持たせるようにしてもよい。1060は実プロセッサ1に組み込まれる障害検出回路、1061は実プロセッサ1に組み込まれる内部情報遮断回路である。本実施例では、障害検出回路101060および内部情報遮断回路1061を、実プロセッサ1の内部にのみ示しているが、実プロセッサ2にも組み込まれるものである。1071は前記実プロセッサへ割込み信号を発生させる割込み回路である。1100は主記憶装置1050内のホストOS記憶領域、1110はゲストOS1の制御情報、1120はゲストOS2の制御情報、1130はゲストOS3の制御情報である。前記ゲストOSの制御情報内に、1111、1112、1121、1131の仮想プロセッサ制御情報が含まれる。仮想プロセッサ2制御情報1112に、1115のゲスト割込みコードが含まれる。仮想プロセッサ1制御情報1121に1125のゲスト割込みコードが含まれる。ここで示したもの以外にも、他の仮想プロセッサ制御情報にもゲスト割込みコードが含まれる。1140はホストOS内の実プロセッサ割当てテーブルである。1150～1152は主記憶装置1050内のゲストOS1～OS3の主記憶領域、1155はゲストOS1に含まれる中断回復

処理であり、1156はゲストOSに含まれる中断回復処理である。他のゲストOSにも中断回復処理は含まれる。1160は割込み情報領域、1161は割込みの内容(例えば、プロセッサ、メモリ、I/O等)を示す識別コード格納領域、1162は障害実プロセッサ番号格納領域である。1170はHSA51内に確保される論理区画計算機遮断領域、1171は論理区画計算機遮断領域内の中断時の情報を格納する領域、1172は論理区画計算機遮断領域内の中断した仮想プロセッサ制御情報アドレスを格納する領域である。1180はHSA51内に確保される実計算機遮断領域、1181は実計算機遮断領域内の実プロセッサ中断時の情報を格納する領域である。

【0039】本発明では、実プロセッサの固定障害により中断した論理区画計算機を回復するため、論理区画計算機のOSに中断回復処理1155、1156等を設け、マルチプロセッサ構成の仮想プロセッサ上で中断処理を救済し、障害状態の仮想プロセッサを切離し、論理区画計算機を救済するところに特徴がある。図16は、実プロセッサに固定障害が発生した時に動作していた処理の状態を遮断する論理区画計算機遮断領域1170である。仮想プロセッサ中断時の情報1171内に、プログラム状態語1310、汎用レジスタの内容1311、浮動小数点レジスタの内容1312、制御レジスタの内容1313、タイマ値1314などが遮断され、1172に仮想プロセッサ制御情報アドレスが遮断される。図17は、実プロセッサ割当てテーブル1140を示し、実施例の論理区画計算機の仮想プロセッサ構成と、仮想プロセッサと実プロセッサとの接続状態を示す表1400であり、実施例の説明のための実例である。ゲストOS1の仮想プロセッサ1とゲストOS2の仮想プロセッサ1が実プロセッサ1、ゲストOS1の仮想プロセッサ2とゲストOS3の仮想プロセッサ1が実プロセッサ2に割当てられ、それぞれオンライン状態(ゲストOSの処理対象状態)で接続されている。

【0040】実プロセッサ1において、固定障害が発生した時の仮想計算機システム回復処理を、以下説明する。実プロセッサ1の内部情報遮断回路1061の動作を、図18を用いて説明する。従来技術で提供されている障害検出回路1060により、実プロセッサ1で発生した固定障害が検出される(ステップ501)。中断した命令の実行内容を保証するため、実プロセッサ1の命令実行マイクロプログラムを停止する(ステップ502)。内部制御プロセッサに固定障害の検出を報告する(ステップ503)。仮想プロセッサの中断時の情報を論理区画計算機遮断領域へ遮断する(ステップ504)。中断した仮想プロセッサの制御情報アドレスを論理区画計算機遮断領域へ遮断する(ステップ505)。仮想プロセッサの状態遮断完了後、実プロセッサの中断情報を遮断するため実計算機モードへ切り替える(506)

6)。実プロセッサの中断時の情報を実計算機退避領域1180へ退避する(507)。そしてチェックストップとする(ステップ508)。

【0041】固定障害の発生を報告された内部制御プロセッサは、割込み回路1071により実プロセッサ2へ割込み信号を発行し、割込み情報1160内の識別コード1161へ実プロセッサ障害を示すコード、障害実プロセッサ番号1162へ実プロセッサ1のプロセッサ番号を格納する。

【0042】実プロセッサ1の固定障害による割込みを受けた実プロセッサ2動作を、図19を用い説明する。実プロセッサ2の上でゲストOS1が走行しているとする(ステップ601)。固定障害による割込みが発生し、実プロセッサの内部制御により、ゲストOS走行モードからホストOS走行モードへ切り替わる。ホストOSは、実プロセッサ障害による割込みを検出する(ステップ602)。割込み情報1160から、障害解析を行なう(ステップ603)。そして、障害実プロセッサを割当てられていたマルチプロセッサ構成のゲストOS1の仮想プロセッサ1制御情報に、ゲスト割込みとしてゲスト割込みコードを格納し、障害実プロセッサを割当てられていたシングルプロセッサ構成のゲストOS2の仮想プロセッサ1制御情報に、ゲスト割込みとしてゲスト割込みコードを格納する(ステップ604)。そして、ホストOSはゲストの入出力のシミュレーション等のゲスト制御を行なう(ステップ605)。その後、実プロセッサの内部制御により、ホストOS走行モードからゲストOS走行モードへ切り替わり、実行待ち状態であったゲストOS3が走行する(ステップ606)。ゲストOS3に割当てられているプロセッサ使用時間が終るか、ホストOSへの割込みが発生したとき、実プロセッサの内部制御により、ゲストOS走行モードからホストOS走行モードへ切り替わる。そして、ホストOSはゲストの入出力のシミュレーション等のゲスト制御を行なう(ステップ607)。その後、実プロセッサの内部制御により、ホストOS走行モードからゲストOS走行モードへ切り替わり、実行待ち状態であったゲストOS1が走行する(ステップ608)。ゲストOS1は、前記実プロセッサの固定障害発生によるゲスト割込みを検出し、実プロセッサ障害による割込みが発生する(ステップ609)。実プロセッサ1の固定障害によるゲストOS1の仮想プロセッサ1上で実行されていた処理を救済するため、ゲストOS1による中断回復処理1155が行なわれる(ステップ610)。中断処理の回復完了後、ホスト割込みを終了し、ゲストOS1が実行再開する(ステップ611)。その後、ゲストOS1に割当てられているプロセッサ使用時間が終るか、ホストOSへの割込みが発生したとき、実プロセッサの内部制御により、ゲストOS走行モードからホストOS走行モードへ切り替わる。

【0043】ゲストOS1で行なわれる中断回復処理1155の動作を、図20を用い説明する。障害仮想プロセッサの制御情報アドレスを、ゲストOSがアクセス可能な主記憶領域へ読み込む(ステップ701)。論理区画計算機退避領域内の中断時の情報を、アクセス可能な主記憶へ読み込む(ステップ702)。退避されている中断時の情報に、中断処理があるか判断する(ステップ703)。中断処理がある場合、中断時の情報による中断処理の回復を行ない(ステップ704)、回復した中断処理を、障害仮想プロセッサの制御情報アドレスの内容に反映させる(ステップ705)。すなわち、内容を更新する。ステップ703により中断処理がないと判断した後、またはステップ705完了後、障害仮想プロセッサで実行待ちとなっている処理のみを正常仮想プロセッサに割当てる(ステップ706)(この場合は、論理区画計算機システムでの例外的処理となる)。そして、この処理が完了した後、障害仮想プロセッサをオフライン状態(ゲストOSの非処理対象状態)とする。なお、実計算機動作の回復を行なうようにしてもよい。この回復処理は実施例1と同様であるので、説明を省略する。

【0044】図21は、本発明を実施し救済した時の論理区画計算機の仮想プロセッサ構成と、仮想プロセッサと実プロセッサとの接続状態を示す実プロセッサ割当てテーブル1140の表1800である。ゲストOS1の仮想プロセッサ1とゲストOS2の仮想プロセッサ1が実プロセッサ1、ゲストOS1の仮想プロセッサ2とゲストOS3の仮想プロセッサ1が実プロセッサ2に割当されている。ゲストOS1の仮想プロセッサ1はオフライン状態、仮想プロセッサ2はオンライン状態であり、ゲストOS2の仮想プロセッサ1は実プロセッサの固定障害によるダウン状態であり、ゲストOS3の仮想プロセッサ1はオンライン状態である。

【0045】障害実プロセッサ1を交換した後のホストOSの処理およびゲストOS2の処理を、図22を用い説明する。保守作業により、障害実プロセッサ1の交換、リセット等が行なわれる(ステップ901)。その後、実プロセッサ1が正常状態となり、ホストOSが走行可能となり、ホストOSはゲストの入出力のシミュレーション等のゲスト制御を行なう(ステップ902)。その後、実プロセッサの内部制御により、ホストOS走行モードからゲストOS走行モードへ切り替わり、実行待ち状態であったゲストOS2が走行する(ステップ903)。ゲストOS2は、前記実プロセッサの固定障害発生によるゲスト割込みを検出し、実プロセッサ障害による割込みが発生する(ステップ904)。実プロセッサ1の固定障害によるゲストOS2の仮想プロセッサ1上で実行されていた処理を救済するため、ゲストOS2による中断回復処理1156が行なわれる(ステップ905)。中断処理の回復完了後、ホスト割込みを終了し、ゲストOS2が実行再開する(ステップ906)

6)。

【0046】ゲストOS2で行なわれる中断回復処理156の動作を、図23を用いて説明する。障害仮想プロセッサの制御情報アドレスを、ゲストOSがアクセス可能な主記憶領域へ読み込む（ステップ951）。論理区画計算機退避領域内の中断時の情報を、アクセス可能な主記憶へ読み込む（ステップ952）。退避されている中断時の情報に、中断処理があるか判断する（ステップ953）。中断処理がある場合、中断時の情報による中断処理の回復を行ない（ステップ954）、回復した中断処理を、障害仮想プロセッサの制御情報アドレスの内容に反映させる（ステップ955）。ステップ953により中断処理がないと判断した後、またはステップ955完了後、処理を終了する。なお、障害によりオンライン状態になっているゲストOS1の仮想プロセッサ1のオンライン状態への変更あるいはオンライン状態の維持についてはゲストOS1が判断・処理するが、本実施例には直接関係がないため、説明は省略する。

【0047】以上述べたように、本実施例によれば、論理区画計算機システムにおいて、実プロセッサ上で固定障害が発生しても、論理区画計算機を停止させることなく動作を継続させる方法を提供できる。

【0048】

【発明の効果】本発明によれば、マルチプロセッサ構成上で稼動する仮想計算機システムにおいて、実プロセッサで固定障害が発生したときの障害回復が可能となる。また、仮想計算機の処理性能を低下させることなく、仮想計算機を救済することが可能となる。また、プロセッサ構成を変更することなく、仮想計算機を救済することが可能となる。さらに、マルチプロセッサ構成上で複数の論理区画計算機が走行する論理区画計算機システムにおいて、固定障害となった実プロセッサが割り当てられた論理区画計算機の仮想プロセッサを中断状態から回復することが可能となる。また、実プロセッサの障害解消時に、実プロセッサが障害のために実行待ち状態であった論理区画計算機の仮想プロセッサの処理を救済することができる。

【図面の簡単な説明】

【図1】本発明の仮想計算機システムの実施例を示す図である。

【図2】従来の仮想計算機システムの例を示す図である。

【図3】仮想計算機退避領域を示す図である。

【図4】実計算機退避領域を示す図である。

【図5】第1の実施例における正常状態における仮想計算機と実プロセッサの割り当て関係を示す図である。

【図6】実プロセッサの内部情報退避回路の処理のフローチャートを示す図である。

【図7】内部制御プロセッサの障害回復回路の処理のフローチャートを示す図である。

【図8】ホストOSの中断ゲスト救済処理における中断仮想プロセッサ回復処理のフローチャートを示す図である。

【図9】ホストOSの中断ゲスト救済処理における重要仮想計算機回復処理のフローチャートを示す図である。

【図10】第1の実施例における障害回復状態における仮想計算機と実プロセッサの割り当て関係を示す図である。

【図11】第2の実施例における正常状態における仮想計算機と実プロセッサの割り当て関係および救済属性を示す図である。

【図12】ホストOSの中断ゲスト救済処理における通常仮想計算機回復処理を示す図である。

【図13】第2の実施例における障害回復状態における仮想計算機と実プロセッサの割り当て関係および救済属性を示す図である。

【図14】実施例3の論理区画計算機システムの構成を示す図である。

【図15】第1の実施例における正常状態における仮想計算機と実プロセッサの割り当て関係を示す図である。従来の論理区画計算機システムの構成例を示す図である。

【図16】論理区画計算機退避領域を示す図である。

【図17】実施例3の正常状態における論理区画計算機と実プロセッサの割り当て関係を示す図である。

【図18】実施例3における実プロセッサの内部情報退避回路の処理のフローチャートを示す図である。

【図19】実施例3における固定障害による割り込みを受けた正常実プロセッサの動作を示す図である。

【図20】ゲストOS1の中断回復処理処理のフローチャートを示す図である。

【図21】実施例3の救済後における論理区画計算機と実プロセッサの割り当て関係を示す図である。

【図22】障害実プロセッサの交換によるホストOSの処理およびゲストOSの処理のフローチャートを示す図である。

【図23】障害実プロセッサの交換後におけるゲストOS2の中断回復処理のフローチャートを示す図である。

【符号の説明】

10、1010 実プロセッサ1

20、1020 実プロセッサ2

30、1030 内部制御プロセッサ

50、1050 主記憶装置

51、1051 ハードウェアの記憶領域（HSA）

40、1040 内部バス

60、1060 障害検出回路

61、1061 内部情報退避回路

62 モードビット

70 障害回復回路

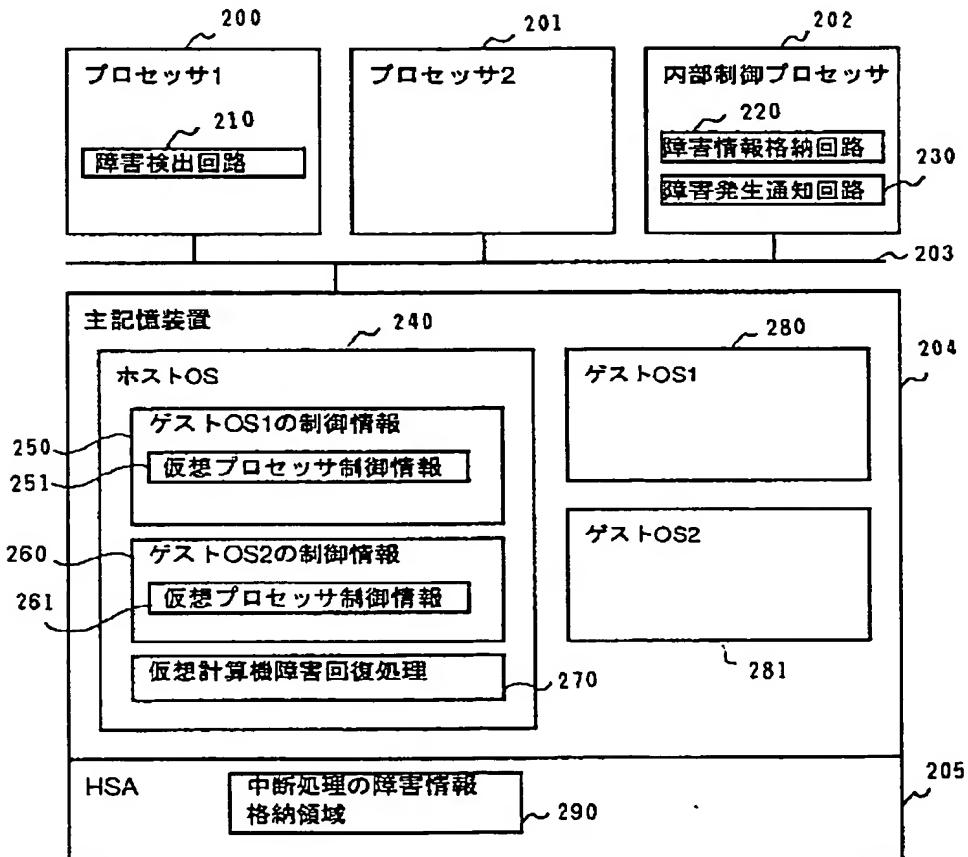
71、1071 割込回路

100 ホストOS記憶領域
 110 ゲストOS1の制御情報
 120 ゲストOS2の制御情報
 130 ゲストOS3の制御情報
 111、121、131 救済属性
 112、122、132 割当て実プロセッサ番号
 113、123、133 仮想プロセッサ制御情報
 140 中断ゲスト救済処理
 150～152 ゲストOS1～OS3の主記憶領域
 160、1160 割込み情報領域
 161、1161 割込み識別コード格納領域
 162、1162 障害実プロセッサ番号格納領域
 170 仮想計算機退避領域
 171 有効フラグ
 172 仮想プロセッサ中断時の情報を格納する領域
 173 中断した仮想プロセッサ制御情報アドレスを格納する領域
 180、1180 実計算機退避領域

181、1181 実プロセッサ中断時の情報を格納する領域
 1100 ホストOS記憶領域
 1110 ゲストOS1の制御情報
 1120 ゲストOS2の制御情報
 1130 ゲストOS3の制御情報
 1111、1121、1131 仮想プロセッサ制御情報
 1115、1125 ゲスト割込みコードを格納する領域
 1140 実プロセッサ割当てテーブル
 1150～1152 ゲストOS1～OS3の主記憶領域
 1155～1156 中断回復処理
 1170 仮想計算機退避領域
 1171 仮想プロセッサ中断時の情報を格納する領域
 1172 中断した仮想プロセッサ制御情報アドレスを格納する領域

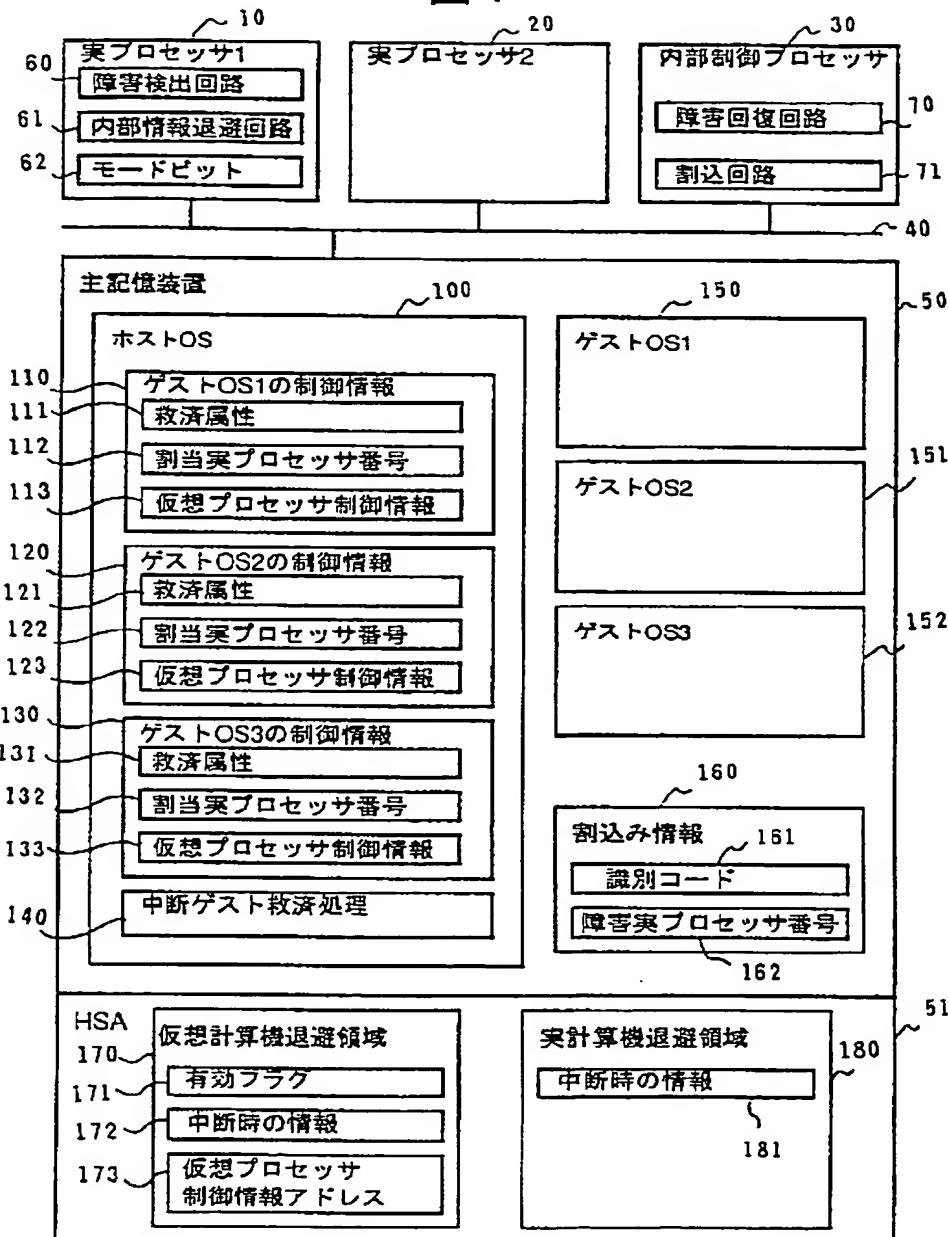
【図2】

図2



【図1】

図1



【図5】

ゲスト	設定内容	割当実プロセッサ番号
ゲストOS1	重基ゲスト	1
ゲストOS2	通常ゲスト	2
ゲストOS3	通常ゲスト	2

図5

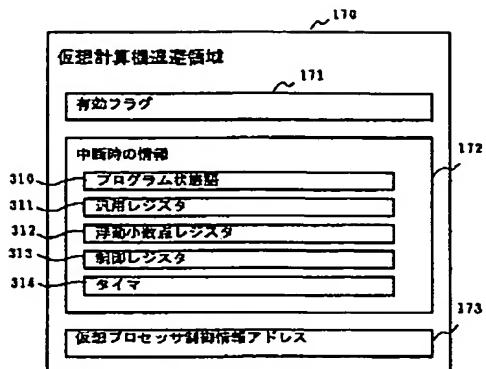
【図10】

図10

ゲスト	設定内容	割当実プロセッサ番号
ゲストOS1	重基ゲスト	2
ゲストOS2	通常ゲスト	なし(停止状態)
ゲストOS3	通常ゲスト	なし(停止状態)

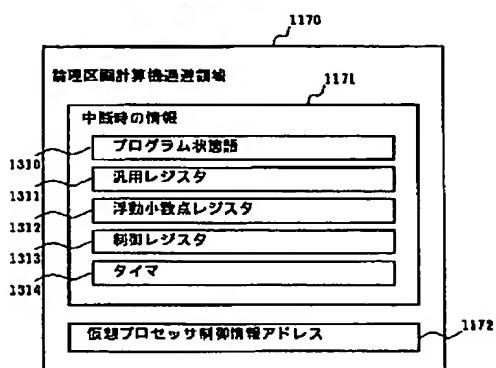
【図3】

図3



【図16】

図16



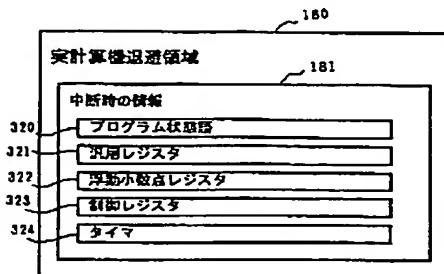
【図21】

図21

ゲスト	実プロセッサ1	実プロセッサ2
ゲストOS1	仮想プロセッサ1 オフライン	仮想プロセッサ2 オンライン
ゲストOS2	仮想プロセッサ1 ダウン状態	—
ゲストOS3	—	仮想プロセッサ1 オンライン

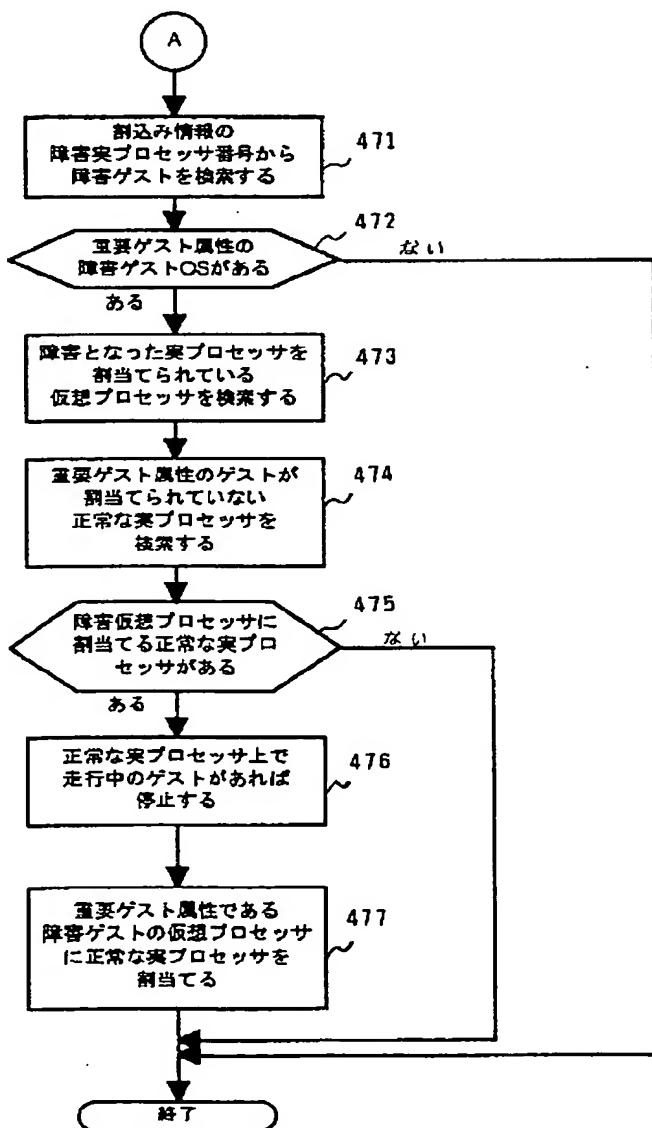
【図4】

図4



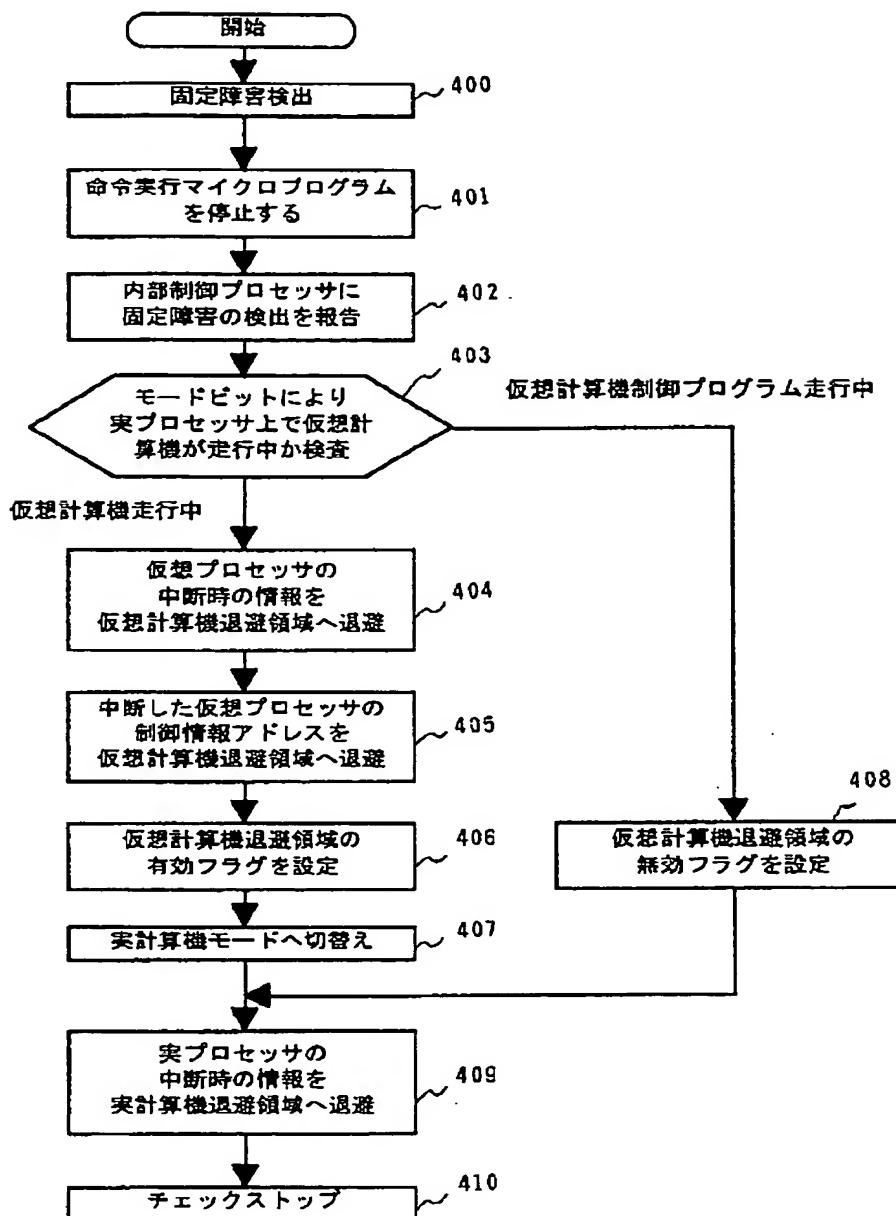
【図9】

図9



【図6】

図6



【図11】

図11

ゲスト	統計属性			
	重要属性	統計時の 実プロセッサ使用比	割当て 実プロセッサ	実プロセッサ 使用比
ゲストOS1	通常	90%	1	90%
ゲストOS2	通常	0%	1	10%
ゲストOS3	通常	20%	2	100%

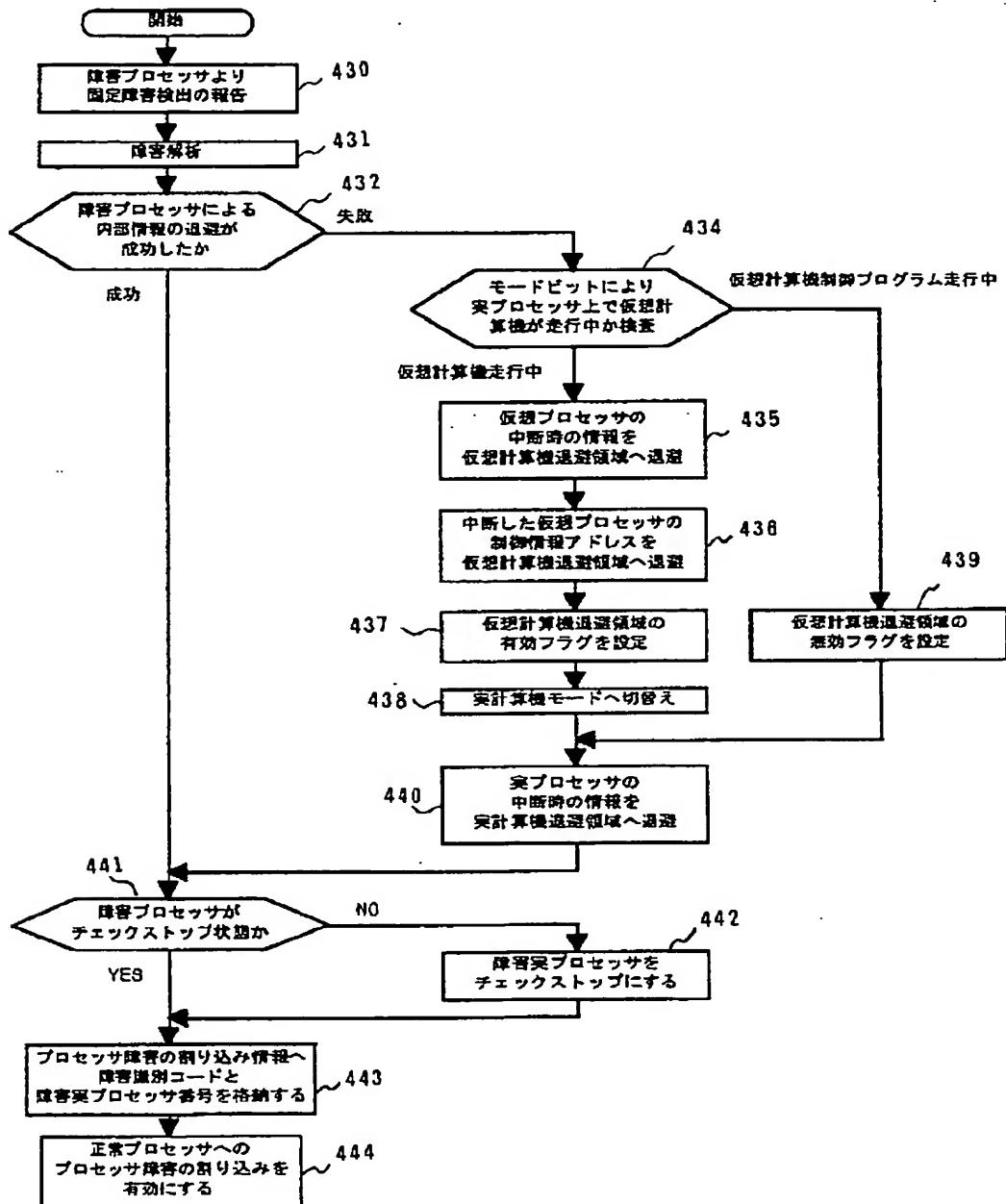
【図13】

図13

ゲスト	統計属性			
	重要属性	統計時の 実プロセッサ使用比	割当て 実プロセッサ	実プロセッサ 使用比
ゲストOS1	通常	90%	2	72%
ゲストOS2	通常	0%	2	8%
ゲストOS3	通常	20%	2	20%

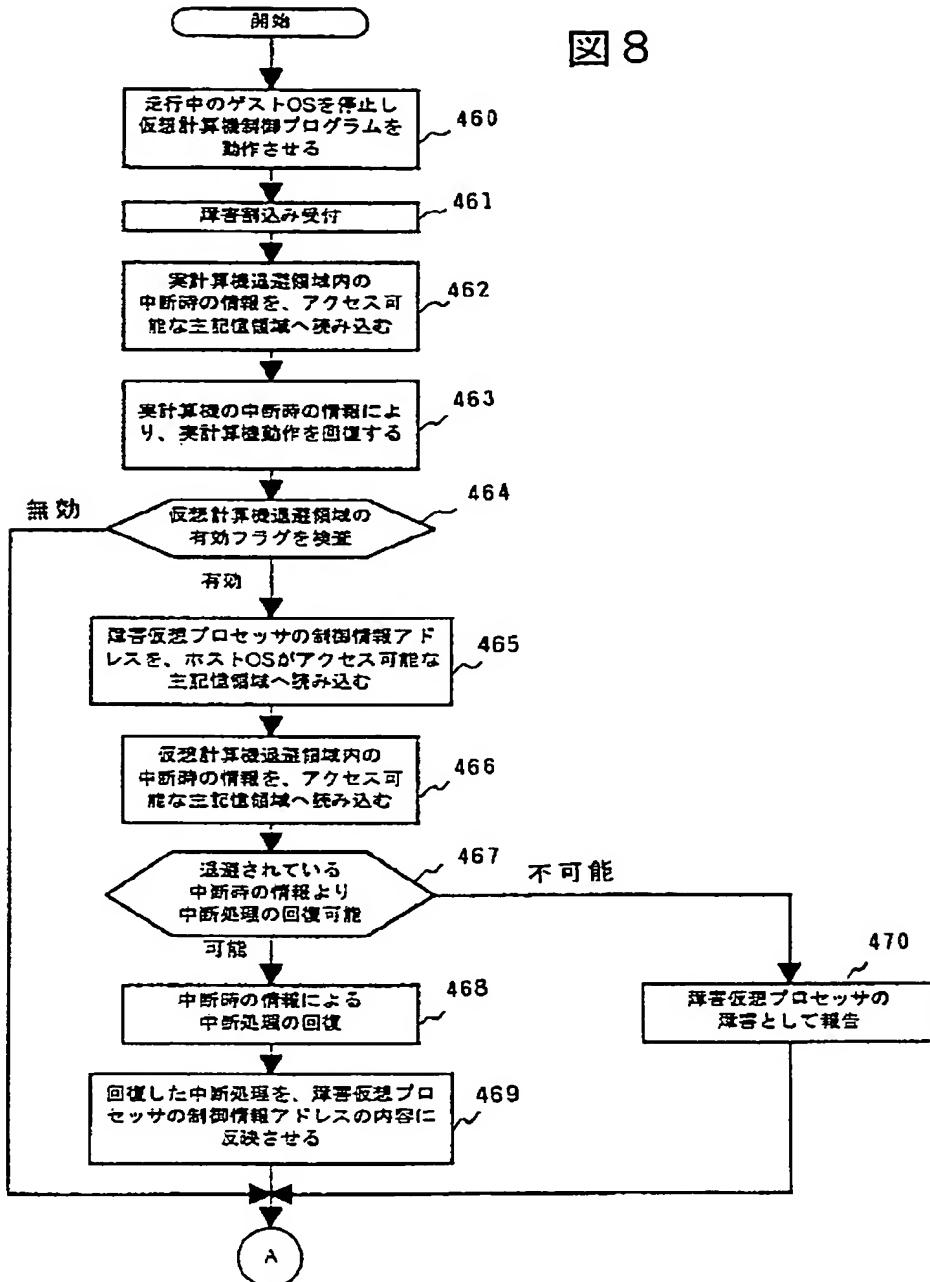
【図7】

図7



【図8】

図8



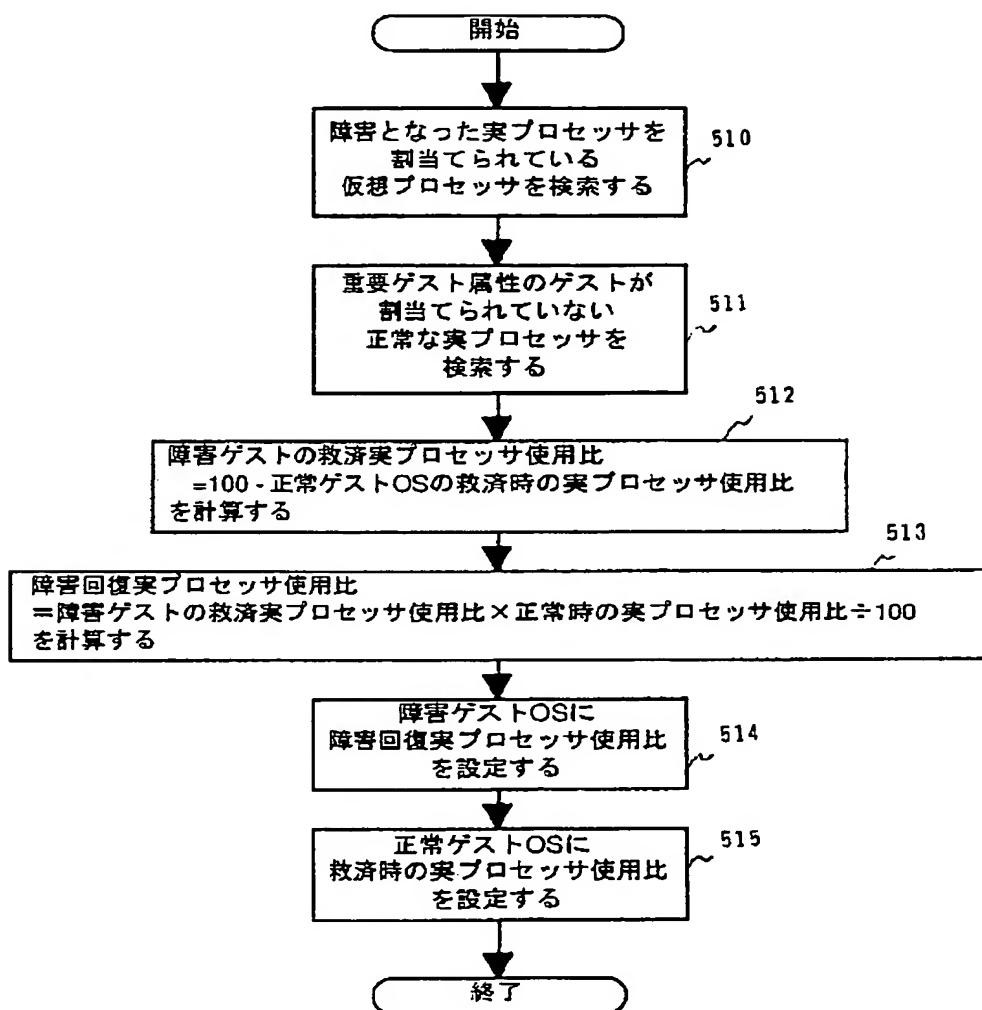
【図17】

図17

ゲスト	実プロセッサ1	実プロセッサ2
ゲストOS1	仮想プロセッサ1 オンライン	仮想プロセッサ2 オンライン
ゲストOS2	仮想プロセッサ1 オンライン	—
ゲストOS3	—	仮想プロセッサ1 オンライン

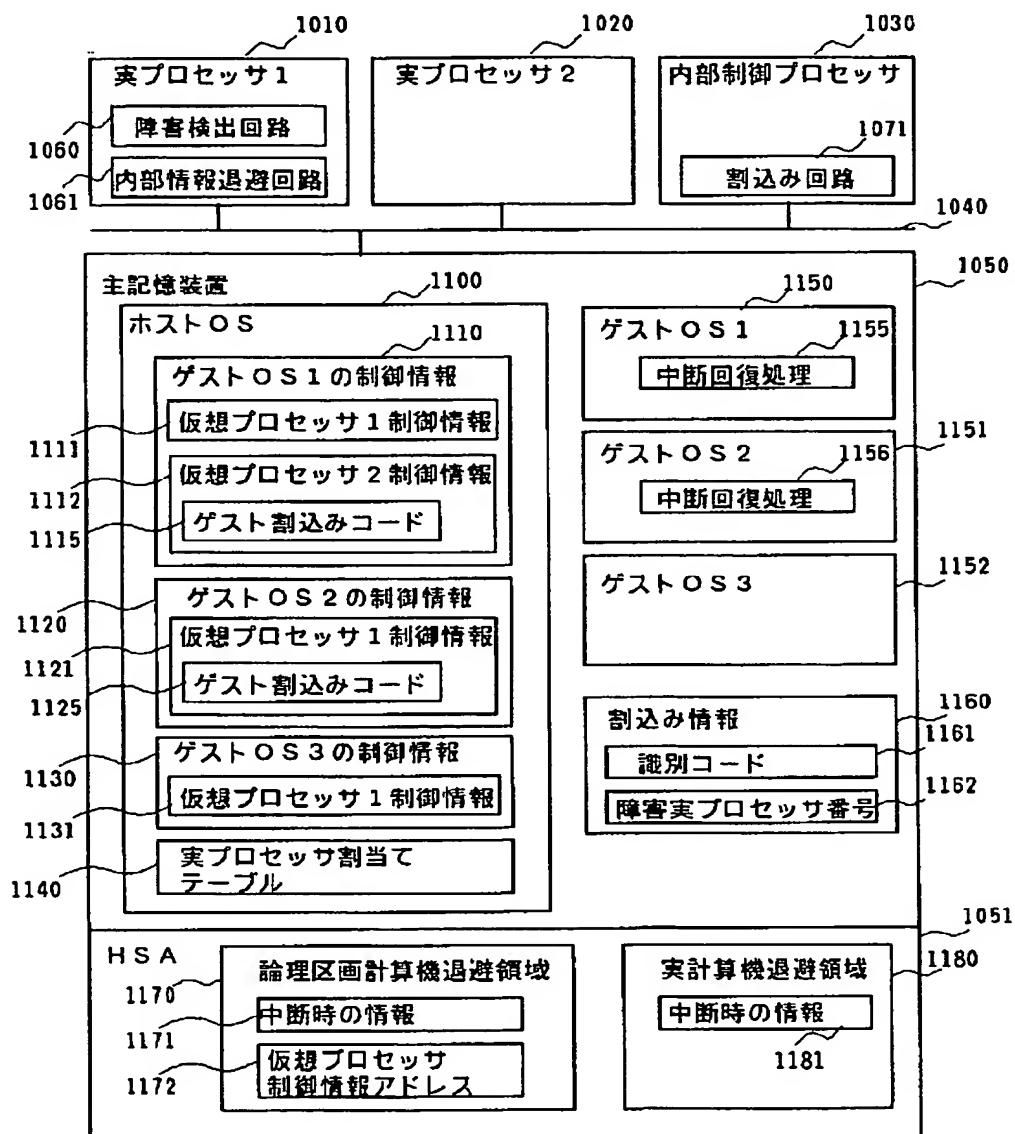
【図12】

図12



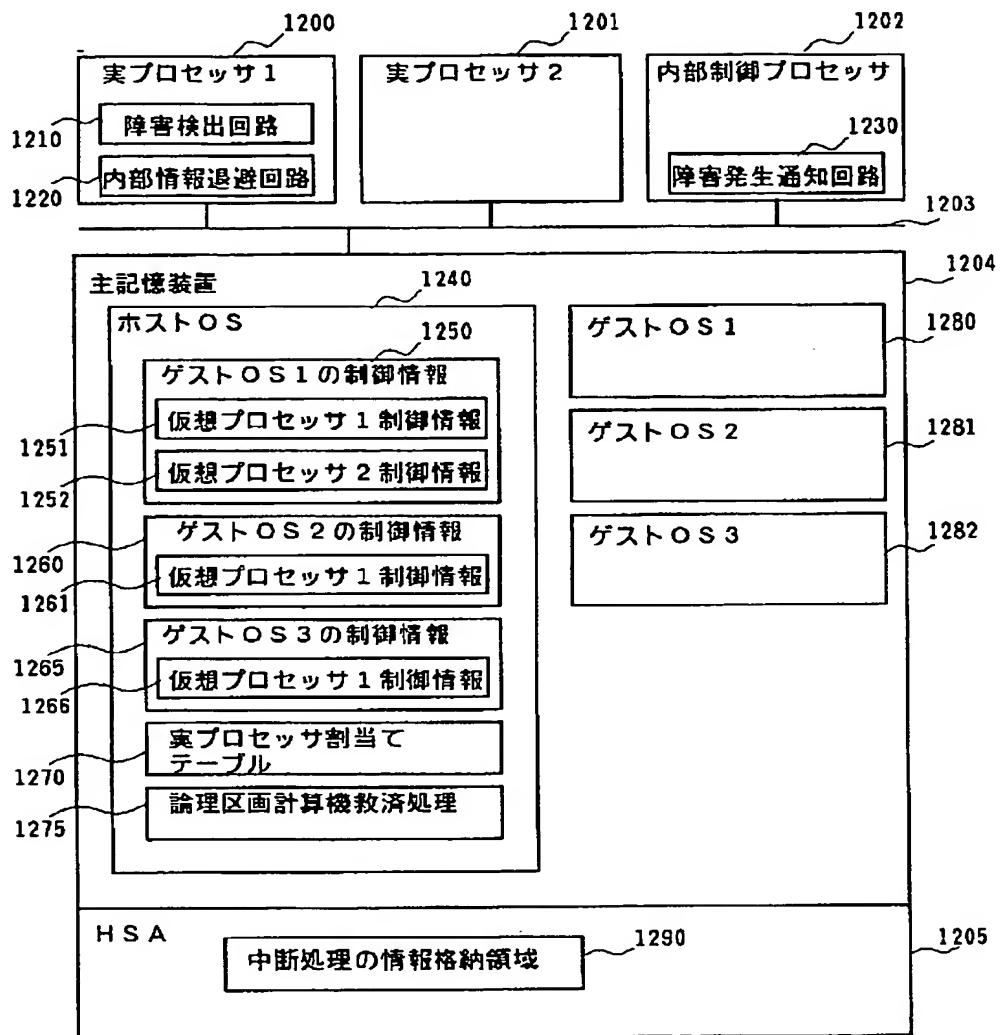
【図14】

図14



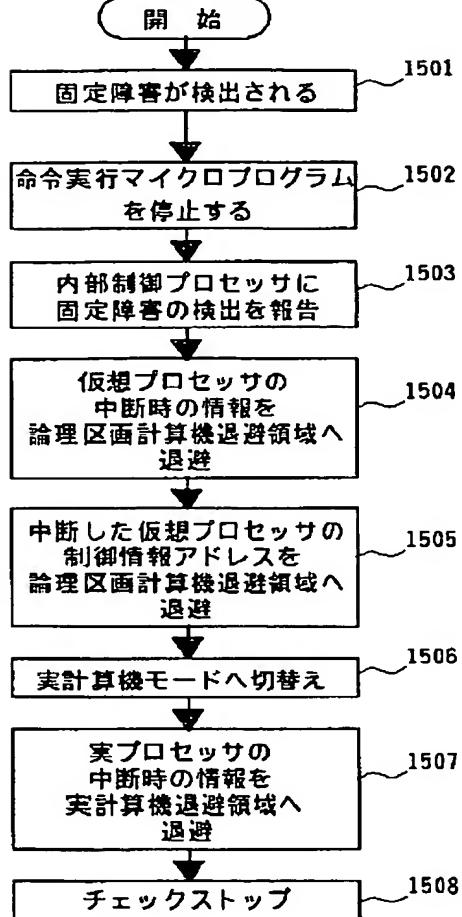
【図15】

図15



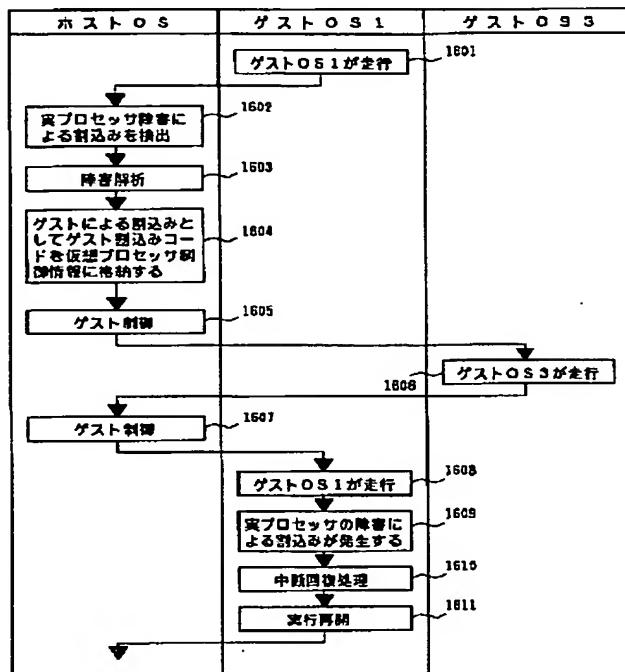
【図18】

図18



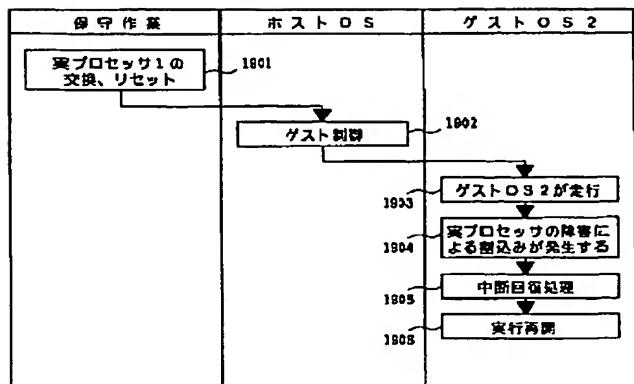
【図19】

図19



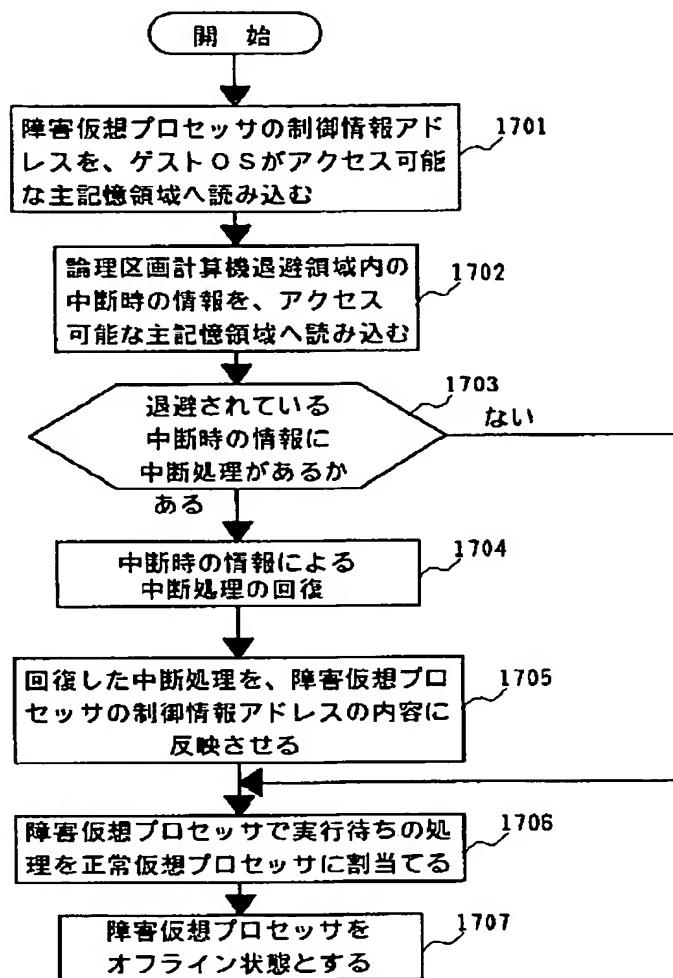
【図22】

図22



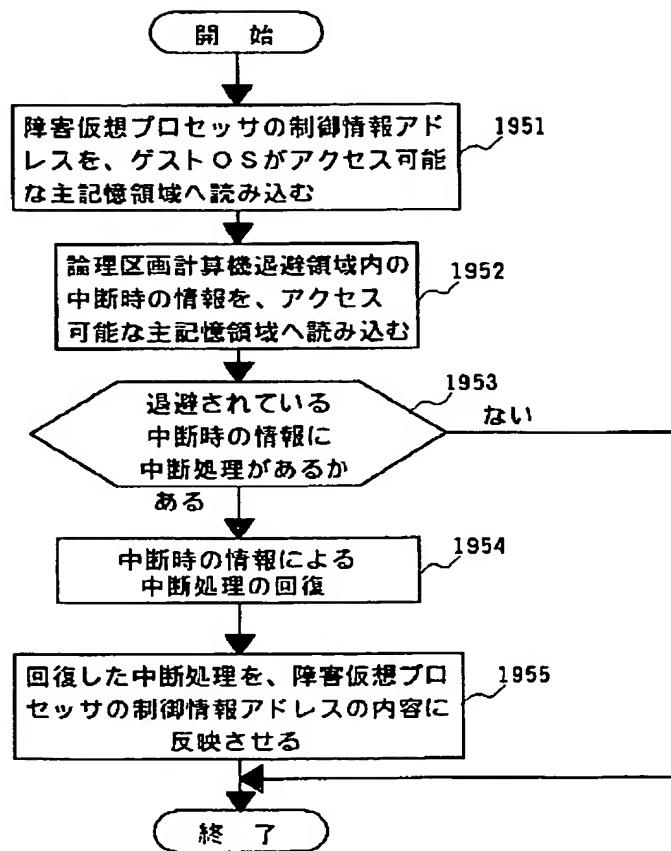
【図20】

図20



【図23】

図23



フロントページの続き

(72)発明者 田中 俊治

神奈川県川崎市麻生区王禅寺1099番地 株
式会社日立製作所システム開発研究所内

(72)発明者 木下 俊之

神奈川県川崎市麻生区王禅寺1099番地 株
式会社日立製作所システム開発研究所内

【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第3区分

【発行日】平成11年(1999)12月24日

【公開番号】特開平8-263454

【公開日】平成8年(1996)10月11日

【年通号数】公開特許公報8-2635

【出願番号】特願平7-90314

【国際特許分類第6版】

G06F 15/16 470

【F I】

G06F 15/16 470 R

【手続補正書】

【提出日】平成11年5月6日

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】 マルチプロセッサ構成の計算機上で、複数の仮想計算機が走行する仮想計算機システムにおける障害回復処理方法において、

実プロセッサに固定障害が発生したとき、該実プロセッサが割り当てられた仮想計算機の状態情報および該状態情報の有効・無効を示す有効フラグと、該実プロセッサの状態情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、

障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、

割り込みを受け付けた正常な実プロセッサは、前記主記憶装置に格納された割込み情報と実計算機退避領域の実プロセッサの状態情報に基づき障害実プロセッサの処理を回復し、かつ前記仮想計算機退避領域の有効フラグが有効を示すとき該退避領域の仮想計算機の状態情報に基づき障害仮想計算機の処理を回復し、

固定障害となった実プロセッサが割り当てられている仮想計算機を検索し、該検索した仮想計算機に与えられた救済属性に基づき該仮想計算機に新たに割り当てられる正常な実プロセッサを選択し、

該正常な実プロセッサにおける仮想プロセッサの割り当て構成を変更することにより実プロセッサの固定障害から仮想計算機システムを救済することを特徴とする仮想計算機システムの障害回復処理方法。

【請求項2】 請求項1記載の障害回復処理方法において、

実プロセッサに固定障害が発生したときに仮想計算機が走行している場合は、仮想計算機の状態情報として中断時の情報および仮想計算機制御アドレスと、該状態情報が有効であることを示す有効フラグを前記仮想計算機退避領域に格納し、

実プロセッサが走行している場合は、仮想計算機の状態情報が無効であることを示す有効フラグを前記仮想計算機退避領域に、該実プロセッサの中断時の情報を前記実計算機退避領域に格納することを特徴とする仮想計算機システムにおける障害回復処理方法。

【請求項3】 マルチプロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおける障害回復処理方法において、

実プロセッサに固定障害が発生したとき、該障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の中断時の情報および仮想計算機制御情報アドレスと、該障害実プロセッサの中断時の情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、

障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、

割り込みを受け付けた正常な実プロセッサは、前記割込み情報に基づき障害解析を行ない、障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の制御情報に該論理区画計算機のゲストOSのゲスト割込みとしてのゲスト割込みコードを格納し、

前記正常な実プロセッサにおいてホストOS走行モードから前記論理区画計算機のゲストOS走行モードに切り替わったとき、該ゲストOSは該ゲストOSに関連する前記ゲスト割込みコードを検出し、該割込みの発生により前記仮想計算機退避領域の情報に基づき中断回復処理を行ない、

該中断回復処理後、前記障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の実行を再開するこ

とを特徴とする論理区画計算機システムにおける障害回復処理方法。

【請求項4】 マルチプロセッサ構成の計算機上で、複数の論理区画計算機が走行する論理区画計算機システムにおける障害回復処理方法において、
実プロセッサに固定障害が発生したとき、該障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の中断時の情報および仮想計算機制御情報アドレスと、該障害実プロセッサの中断時の情報を夫々主記憶装置の仮想計算機退避領域と実計算機退避領域に格納し、
障害の発生を正常な実プロセッサに報告し、該正常な実プロセッサは発生した障害を解析して障害実プロセッサ番号と障害の内容を示す識別コードを主記憶装置へ割込み情報として格納し、該正常な実プロセッサまたは他の正常な実プロセッサに割り込みを行ない、
割り込みを受け付けた正常な実プロセッサは、前記割込

み情報に基づき障害解析を行ない、障害実プロセッサが割り当てられた論理区画計算機内の仮想計算機の制御情報に該論理区画計算機のゲストOSのゲスト割込みとしてのゲスト割込みコードを格納し、
障害実プロセッサの障害が解消され、障害が解消された実プロセッサにおいてホストOS走行モードから実行待ち状態にある前記論理区画計算機のゲストOS走行モードに切り替わったとき、該ゲストOSは該ゲストOSに関連する前記ゲスト割込みコードを検出し、該割込みの発生により前記仮想計算機退避領域の情報に基づき中断回復処理を行ない、
該中断回復処理後、前記障害実プロセッサが割り当てられた実行待ち状態にある論理区画計算機内の仮想計算機の実行を再開することを特徴とする論理区画計算機システムにおける障害回復処理方法。

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.